I

I Interrupt disable flag of 6502, 312
.I SUPERMON single-step, 300, =.W in EXTRAMON, 301 / Integrate memory, 302
IBM 132, 151, 158, 165
I.D. of Disk see HEADER. 230
IEEE bus: 374, 375, 378 / References, 376 / CBM version: Port, 3, 374 / Machine-code examples, 376, 377, 379 - 382 / Handshaking and ATN, 375, 379 [2], 380 [4] / Logical files, device numbers, secondary addresses, see Parameters / Examples include GET#, 69 [3], SAVE, 127, ST, 139, 381 [5]
IF (BASIC) next line if false, THEN or GOTO if true, 75
?ILLEGAL DIRECT ERROR eg. INPUT, 76
?ILLEGAL QUANTITY ERROR examples include LOG, SQR with -ve argument, and ASC, MID$ with string of length zero
INC (6502 opcode) increment address, 335
INITIALISE (BASIC < 4 disk) 166, 231
INPUT (BASIC) 76 / Crashproofing, 25, 77, 254 / Use of GET, 26-28 / Standard Data Entry Environment, 476
Input buffer Position, description, watching it in use, 6, 16, 79, 351 / Example, MERGE 93 - 94 / see BASIC
Input file, current eg. 77 [2], 93
INPUT#(BASIC) 78, 114, 163, 169
INSTRING$ inserts a BASIC string, 80
INT (BASIC) rounds down, 81
Interactive System 21
INX (6502 opcode) increments X-register 335
INY (6502 opcode) increments Y-register 336
Interrupt see VIA, IRQ, and NMI
Interpolation, Inverse 446, 455, 456
IPUG (or ICPUG) 2, 93, 120, 148, 195, 226, 243, 439, 488
IRQ vector, 440, 351; generated by screen refresh, registered by PIA location E813, 384 - 385; frequency, 16, 148, 255 / Uses and examples: 257 ff; tunes, 281; graphics, 277; display bytes, 351; keyboard, clock and Stop key, 255, 256; new keyboard, single-key BASIC entry, 259 - 261; Software uncrash, 262
Isaacson, D. 237
Iteration example, 138

J

Jackson, M. 20
Jiffy Clock 147, 148, 157; and see IRQ
JMP (6502 opcode) jumps to new address, 336; Indirect jump bug, 488; other methods, see RTI and RTS, eg. 144 [1]
de Jong, M. 261
JSR (6502 opcode) jump, saving return, 337; popping return address, PLA 342
Jump Tables see eg. 144 [4], 319

K

K Kilobytes, table 296, 484
Kernel CBM standard machine-code jump table. BASIC <4 and BASIC 4, 440
Keyboard 253 - 261; decoding tables, 427, 428; see IRQ, WAIT 157, PIA 255, 383ff, Stop disable 254 / Keyboard buffer, 16, 257; GET (kernel = FFE4) fetches, 68, 254; Examples switching direct mode into program mode, 28, 45, 57, 77, 157, 257, 259 - 260; Exists, 68 [1]
Keywords, BASIC table, 6; see BASIC
Kilobaud-Microcomputing magazine, 1, 93, 159, 176, 198, 235, 262, 375
KIM 3, 264
Knuth, D. 132, 449, 450
Kolbe, W. 244
Kraft, P. 2

L

.L MLM Load command, 238, 298
Lake, M. 132
Languages 494
LDA (6502 opcode) loads accumulator, 338
LDX (6502 opcode) loads X-register, 338
LDY (6502 opcode) loads Y-register, 339
Least-squares methods 462 - 463, 471 - 472
LEFT$ (BASIC) left substring, 82
LEN (BASIC) length of string, 83
Leon, R. 169
LET (BASIC) assignment, 84, 155
Levinson, F. 61
Linefeed CHR$(10) eg. 79 [2]; see crlf
Link Address Pointers see BASIC
Lissajou figures example, 129
LIST (BASIC) 5, 86; ROM differences, 87; Printer lower-case, 251; Lists screen edit characters, 88, 357 - 358; and TRACE, 149
Literals 11
Liverpool Software Gazette 108, 151, 193
LOAD (BASIC) 89 / Pattern matching with * and ?, 89 / Program mode examples, program length, string, and FN DEF bug, use of OLD, 85, 90, 125 [5], 151
?LOAD ERROR 90
Loaders ordinary, 370; relocating, 371
Locksmith 175 - 176
LOG (BASIC) converse of exp, 91
Logical expressions 12 et al.
Logical file number 90, 127; see parameters
LOMEM & HIMEM alter BASIC RAM available to a program 92
Lookup Tables 33
Loops see FOR...NEXT, 64 - 66, 18 / exit and nesting, 66, 98 / with IF, 64
LSR (6502 opcode) shift right, 339

M

.M MLM command to display memory, 297
MACHINE-CODE: The 6502 chip, Chap. 11, Addressing 310; Flags 312; Program counter, stack 313; Hardware vectors 313; instructions 314; Tables appendices Programming, Debugging 373; Opcodes Chapter 12; Program methods, 315 ff in Chapter 11 / with BASIC, 28, 365 ff / Disks, 194 ff / Graphics, 276 ff / Keyboard, 255 ff / Mathematics, 465 ff / Screen, 6 introductory elementary programs, 307 - 309
And ROMs, BASIC operation, Chap. 5 / Index to ROM routines, Chapter 15 / Selected examples, Chapter 13 / Conversion between ROMs, 364
Maclean, W. 80
Macro Feature of some assemblers, 364
Malmsberg, D. 278
Mask see eg. 324 [1], 340 [1]
Matrix Definitions, examples 458 ff
Maynard, M. 236
McCracken, W. 243
M-E (CBM disk) memory execute, 191
Mead, T. 108
Memfix shifts BASIC RAM, 14, 92
Memory Map RAM and ROM, Chapter 15
Memory Move 64, 301, 355
Menu 21, 474, 476
MERGE Interconnects BASIC programs, Tape 93; Disk 94
Micro Magazine for 6502 and 6809, 61, 120, 128, 245, 251
Micropolis disk drives, 3, 158, 213
Microsoft Software writers, 5, 8, 9, 57, 58, 110, 125, 153, 405, 462, 464
MID$ (BASIC) takes substring, 95, 82 [2]
Midnite Software Gazette 251
Mikro Assembler chip, 362
MLM (Machine Language Monitor) 296 ff; Operation and subroutines, 418, 420; Extended monitors: see Monitor
MMF (Micro-Mainframe) 1, 307
MOD calculates remainders, 96
Modem 253
Modes: (i) Direct and Program, eg. 6, 152 [6], Chapters 5 and 7 for individual keywords / (ii) Lower-case and Graphics, screen modes, eg. 5, 11, 268, 272
Molloy, J. 2
Monitor for machine-code, see MLM; In BASIC, 304 ff; Extended monitors, 298 - 299, and see EXTRAMON and SUPERMON
MOS Technology part of Commodore Semiconductor Group, 1
MPI disk drives, 198
M-R (CBM disk) memory read, 189
Mu PET multi user disk system, 159
MUSE (Education users), 19, 481
M-W (CBM disk) memory write, 190

N

N Flag for bit 7, 312
.N Extramon 'New Locater' command, 302
Nassi-Schneiderman chart, 19
NEW (BASIC) sets BASIC pointers to start position, 97 / Disk NEW, see HEADER
Newman & Sproull 282
NEXT (BASIC) returns to start of previous FOR statement, held on stack, 98
?NEXT WITHOUT FOR ERROR 66 [6]
NMI (non-maskable interrupt), 262, 313, 440
Normal Distribution 449 - 450
NOP (6502 opcode) no operation, 340
NOT (BASIC) unary 16-bit operator, 99
Numerals: see Accuracy, Calculations, Floating-Point Accumulators, Rounding, Variables/ String interconversion, see STR$ and VAL, and table, 356
Nybble 294; interchange, eg. 364

O

OLD Positions pointers to correspond with BASIC in RAM, 100 & 90 [2], 224 [2]
ON (BASIC) Casentry-like construction, with GOTO or GOSUB, 102
Opcodes 314; Alphabetic list with full details, Chapter 12; 482, 483, 485
OPEN (BASIC) Sets up file-table entries, checks IEEE device, 103; file-table parameters, 104; disassembly, 352
Operators 11; priority, 37
OR (BASIC) 16-bit logical operator, 105
ORA (6502 opcode) 8-bit operator, 340
.ORG Assembler directive (origin), 363
Osborne & Donahue 2, 163
?OUT OF DATA ERROR and READY, 118
?OUT OF MEMORY ERROR 36 / Calculations using stack, eg. 15 / Pointers inconsistent, eg. 48 / Stack depth full, eg. 72 / Missing null byte, eg. 97
OZZ (Database system) 184, 185

P

.P Printer disassembly, 301
Packages 475
Parameters for logical files, eg. 104 [2]
Parametric coordinates eg. 131
Pascal (Computer language) 494
Password 354
Pattern matching with CBM disks, 89
Pause 352
Peddle, C. 2
PEEK (BASIC) 106 / BASIC 1, 92, 153 [1]
Personal Computer World 1
Portec disk drives 198
PET (Personal Electronic Transactor), and CBM passim
Peterson, T. 179
PHA (6502 opcode) Push A 'on stack', 341
PHP (6502 opcode) Push PSR on stack, 341

rry flag of 6502, *312*
xtended monitor branch calc'n, *300*
ılations in machine-code; using ROM, *ap. 16* / without ROM, *Chap. 11*
ılus diff'l, e.g. *63* / integral, e.g. *447*
bell, G. *236*
iage return CHR$(13) see Crlf; as
ord separator &c. e.g. *162-3, 113 [2]*
ntry flowchart structure, *18, 102*
ALOG (BASIC 4 disk) =DIRECTORY
Commodore Business Machines, *passim*
(Commodore Club Newsletter) *2, 132, 7, 193, 194, 235, 251, 253, 278*
ıberlin, H. *289*
ınel to disk, *185* /see Secondary Addr.
ınel Data Book *2*
acter set Screen, *266* /screen RAM, *8* /generator ROM, 267, 272
kdigits, letters e.g. *22-24, 96*
, C. *271*
well, R. *267*
v, H. *41*
$ (BASIC) opp. of ASC, *46,* & eg *240*
GET *365* and eg GOTO, GOSUB, VAL
(6502 opcode) clear carry flag, *330*
(6502 opcode) clear decimal flag, *330*
(6502 opcode) clear int.disable, *330*
k example *274*/ see TI and TI$
SE (BASIC) *47*/ and disk files, eg *211*
(BASIC) Reset variables' pointers, *48*
(6502 opcode) clear V flag, *330*
(BASIC) print, but leave file open
d listening, *49, 77 [2], 113 [1], 379 [1]*
(6502 opcode) compare with A, *331*
OL formatting *115, 162*
es on/off bits within bytes, *24*
LECT (BASIC 4 disk), *219* /=VALIDATE
n statement separator (exc. REM, *119*)
LM function, *297*
IAL 'Common Algorithmic Language' *494*
binatorics *449*
ma data separator eg. *54 [3], 69 [4], 112*
mand-O BASIC 4 EPROM, *115, 263*
modore Business Machines Sources of formation, *1-2*/ Hardware evolution,
omputers *3* / disks *165-7* / tape *235*
pilers *50, 494*
plement NOT, 99/ 2's comp. eg. *112, 295*
pu/think *90, 126, 132, 148, 152, 198-210*
pute! magazine *2, 41, 81, 100, 193, 230, 36, 237, 278, 282, 379*
puter publications *2, 264, 451*
puterist's Guide *2*
NCAT (BASIC 4 disk) *220* /see COPY
catenation of strings with '+' eg. *82*
o, D. *193, 230*
ic sections demonstration *279*
NT (BASIC) Continue program, *50*
tracts, software *478*
ventions in this book *1, 214, 322*
ke, J. *380*

COPY (BASIC 4 disk) *222* / See also *193*
Copy protection, piracy *242, 479*
COS (BASIC) cosine function, *51*
Council for Educational Technology *481*
CP/M 'control program for micros' *263*
CPUCN *2, 80, 176, 243, 264, 265, 278, 490*
CPX (6502 opcode) compare data with X, *332*
CPY (6502 opcode) compare data with Y, *332*
Creative Computing *1, 87, 121, 267, 282*
Crlf Carriage return and line feed, eg. *79 [2], 112 [1], 171, 172*
Crozier, P. *474*
CRT controller chip *270*
CRUNCH BASIC compression, *52*
Cursor RAM locations *156*/ example *45*

D

D Decimal flag of 6502, *312*
.D Extended monitor disassembly, *300*
D-A Conversion Digital to analog, *264, 293*
DATA (BASIC) *54* /examples *24*
Data Compression *22, 28, 29*
Date Processing *24, 25, 132*
David, D. *93*
Davis, R. *194*
DBL Extended precision calculation, *55*
DCLOSE (BASIC 4 disk) writes BAM and updates directory on file close, *223*
Debugging BASIC *36*, IEEE *382 [2]*, machine-code *373*
DEC (6502 opcode) decrement address, *333*
Decimal to hex conversion *23, 295, 465, 484*
DEEK double-byte PEEK, *10*
DEF FN (BASIC) *9, 56*
DEL deletes BASIC lines, *57*
Delay loops *64, 98, 124, 147*
Deleting disk files see SCRATCH
?DEVICE NOT PRESENT ERROR *90, 126, 140*
Device numbers see eg. SAVE
DEX (6502 opcode) decrement X-register *333*
DEY (6502 opcode) decrement Y-register *333*
'Diagnostic Sense' Pin *222*
DIM (BASIC) *58-59*/default, assignment *85*
Direct Mode see eg. INPUT, GET, *77 [5]*
DIRECTORY (BASIC 4 disk) *217-8, 363*
DISKS: Channel 15, *168* Disk Drives, *31*
DOS and ROMs, *159, 193*; Hardware description, diagrams, etc. *158 ff*; reliability and maintenance, *210-213*
Diskettes = floppy disks, description *160ff*
Formatting new diskettes, *168*
?DISK FULL ERROR note *212, 222*
Disk storage Capacity *167*/ Directory *160* / Header and Block Availability Map (BAM) *177-184* / Examples *180-184*
Machine code programming, *194-198*
Pattern Matching with * and ?, *89*
Diskmon Compu/think DOS, *198-210*
Disk o Pro *87, 115, 120, 263*
DLOAD (BASIC 4 disk) *224*
Documentation *119, 478*
DOKE double-byte POKE, *107 [2]*

Dongle hardware security device, *151, 479*
DOPEN (BASIC 4 disk) *225*
DOS 1+, 2+, 2.7, *159, 166, 213* /summary of bugs *211-212* / DOS Support *169*
Double-density graphics see SET
Downey, J. & Rogers, S. *264*
Dr. Dobbs Journal *1*
DS, DS$ *90, 168, 170, 197, 227* / Table *228*
DSAVE (BASIC 4 disk) *229*
DUM disk utility maintenance, *169*
Dummy variable see eg. FRE, POS, DEF FN
DUMP *60, 137, 281-282*
DUPLICATE see BACKUP
Dvorak keyboard *257*

E

.E EXTRAMON command, sets IRQ, *301*
Education *479 ff*
END (BASIC) *62*
EOR (6502 opcode) 8-bit exclusive or, *334*
EPROMs *263*
Erase disk files see SCRATCH
Error messages *25, 32, 71*
Exchange Sort *133*
EXEC File *93*
EXP (BASIC) e^x, converse of LOG, *63*
Expressions, string and numeric eg. *12, 110*
Extension TV for PET/CBM *265*
?EXTRA IGNORED *68, 77* / suppressed *78*
EXTRAMON Extended monitor *301, 177, 298*
Commands ABDEFHINQTUW', *301 ff*
Evans, C. *2*

F

.F Extended monitor 'fill' command, *300*
Factorial example, *63* / formula *449*
Falkner, K. *236*
False value 0, eg. IF *75*
FILES: Description *20, 162*; opening and closing, *166, 226 [1]*; setting up, *226 [2]*; to keyboard, screen, *77 [2], 78, 79*; use of GET#, INPUT#, PRINT#, *169 ff*; BASIC 4 syntax, table of L & W, *223*
?FILE DATA ERROR *77 [2], 78 ii*
?FILE NOT FOUND ERROR *90, 139, 222 [2]*
?FILE TYPE MISMATCH ERROR eg. *90*
File numbers, logical file #, see parameters
File number, active see INPUT, TAB(, SPC(
File types: Direct track & sector, *166, 185-192* Indexed sequential, *165* Inverted, *165* Program, *175* Random Access (indexed with algorithm, + spill procedure) *164* Relative, *163*, examples *173-174* Sequential, *162, 168*, examples *171-172* User, *168*
Files: Disk: Direct track & sector, *185-192* Program, *175* Relative, *173-174* Sequential, *171-172* Compu/think, comparison table, *200*
Files: Tape: Program, *237 ff*
Sequential, example, headers, blocks, *239 ff*

FIND *137*
Finn, K. *61*
Fisher & Jensen *376, 380, 382*
Flags 6502, see processor status register
Floating-point Accumulators *442-443, 464*
ROM with FPAcc#1, table, *466*
ROM with FPAcc#1 & FPAcc#2, table, *467*
Examples, *468 ff* / USR to display, *153* / other examples of use, *99, 115, 131*
Floppy Disks *158, 160*
FN (BASIC) signals function, eg. *10, 12, 56*
FOR..TO..[STEP] (BASIC) loop, *64-66*
Formatting *115, 142, 248 ff*
?FORMULA TOO COMPLEX ERROR *142*
Forrester, J. *451*
FORTRAN *102, 130, 162*
Foster, C. *264*
FRE (BASIC) measures free RAM available to BASIC, *67* / time taken, *67 [3]*
Freeman, R. *159, 198*

G

.G MLM command, Go Run, *298, 419*
Garbage collection *67 [3]*
GET, GET# *68, 69 [4], 169, 352*
GETCHR see BASIC/ put into RAM by RESET, *439* / set to start by CLR, RUN
Glitch, disk *158*
GO (BASIC >1) *16, 70*
GOSUB (BASIC) *71, 72 [1]* /computed, *368*
GOTO (BASIC) *73* /computed, *367*
Grad measurement of angle, *146*
Grainger, B. *488, 489*
Graphics PRINT *110*; Reverse, *111 [2]*; Example, *272*; dumping screen to printer, *281-282*.
Machine-code: double-density, SET, *128, 129, 280* / other, *276 ff* / columns, *278* / table, grouped by type, *273*
Green, N. *111*

H

.H Supermon's Hunt, *300* /modified *303*
Hampshire, N. *2, 132, 149, 265*
Hand Assembly *372*
Handshaking with IEEE, *380 [4]*
Hard Coding *25*
Hardware bugs *36*
Hardware vectors in 6502, see NMI, RESET, IRQ, eg. *440*
Hashtotals BASIC utility example, *369*
HEADER (BASIC 4 disk) *230* / two types, *230 [2]* / = Disk NEW
HELP with syntax, *50* /with system, *476*
Hexadecimal Notation *294 295*
Hex to decimal conversion *23, 295, 465, 484*
Hierarchy of operators, e.g. *37, 99 [3]*
Higginbottom, P. *476*
HTAB horizontal tabulation, *74*

```
LINE# LOC   CODE        LINE



0004  0000              ;**************************************
0005  0000              ;*                                    *
0006  0000              ;*  8050 CONTROLLER FOR MICROP DRIVE  *
0007  0000              ;*                                    *
0008  0000              ;**************************************
0009  0000                     .END
0010  0000                     .LIB DEFS
```

```
E# LOC    CODE        LINE


   0000               ;I/O REGISTERS
   0000               ;
   0000                     * =$40
   0040               ;MOS 6522
   0040               VIAA
   0040               VB    *=*+1         ;PORT B
   0041                                  ;BIT 7 SYNC DETECT 1-NO 0-YES
   0041                                  ;BIT 6 PLL CONTROL BIT
   0041                                  ;BIT 5 MOTOR 0 OFF
   0041                                  ;BIT 4 MOTOR 1 OFF
   0041                                  ;BITS 3-2 STEPPER MOTOR DRIVE #0
   0041                                  ;BITS 1-0 STEPPER MOTOR DRIVE #1
   0041               DIN   *=*+1         ;PORT A
   0042               VDDRB *=*+1         ;DATA DIRECTION REGISTER B
   0043                     * =*+1
   0044               T1LL  *=*+1         ;TIMER 1 LATCH & COUNTER LOW
   0045               TIMER *=*+1         ;TIMER 1 COUNTER HIGH
   0046               T1L   =T1LL
   0046               T1H   =TIMER
   0046                     * =*+5
   004B               ACR   *=*+1         ;AUX CONTROL REGISTER
   004C               PCR   *=*+1         ;PERIPH CONTROL REGISTE
   004D                                  ;BIT 0:    (SET TO 0)
   004D                                  ;CA1: BYTE READY 1:YES 0:NO
   004D                                  ;BITS 1-3:
   004D                                  ;CA2: FILL/SYNC
   004D                                  ;  NORMAL: XC
   004D                                  ;  SYNC/FILL: XE
   004D                                  ;BIT 4:  (SET TO 1)
   004D                                  ;CB1: ERROR DETECTED 1:YES 0:NO
   004D                                  ;BITS 5-7:
   004D                                  ;CB2: READ/WRITE
   004D                                  ;  WRITE: DX
   004D                                  ;  READ: FX
   004D               IFR   *=*+1         ;INT FLAG REGISTER
   004E               IER   *=*+1         ;INT ENABLE REGISTER
   004F               ;
   004F                     *=$80
   0080               ;MOS 6530
   0080               MITA
   0080               DOUT  *=*+1         ;PORT A
   0081               EOUT  *=*+1         ;DIR A
   0082               PB    *=*+1         ;PORT B
   0083                                  ;BIT 6 *** UNUSED ***
   0083                                  ;BIT 4 ODD HEAD SELECT
   0083                                  ;BIT 3 WRITE PROTECT 1:YES
   0083                                  ;BITS 2-1 FREQUENCY (BIT DENSITY)
   0083                                  ;BIT 0 SWITCH 0: DRIVE #0
   0083                                  ;             1: DRIVE #1
   0083               DDRB  *=*+1         ;DATA DIRECT REG B
   0084                     * =*+11
   008F               MITAT *=*+1         ;TIMER/1024
```

```
LOC    CODE           LINE

0090                  ;
0090                  ;RAM ON MOS 6530
0090                  ;0000-003F  0100-013F
0090                  ;
0090                      * =0
0000                1 CLOCK   *=*+1         ; CONTROLLER CLOCK
0001                2 MTRCLK  *=*+1         ; MOTOR CLOCK CLOCK/16
0002                3 MTRTM   *=*+2         ;MOTOR TIMER
0004                                       ;+ WHEN MOTOR FULLY ON
0004                                       ;O WHEN SHOULD TURN MOTOR OFF
0004                5 DRVST   *=*+2         ;DRIVE STATUS WORDS
0006                                       ;BIT 7 ACCELERATING 0:NO 1:YES
0006                                       ;BIT 6 STEPPING 0:NO 1:YES
0006                                       ;BITS 5-0 TRACK#
0006                7 STEPS   *=*+2         ;#STEPS TO NEW TRACK
0008                9 COW     *=*+1         ;USED WITH INTERRUPT
0009              10 WORK    *=*+2
000B     HEAD                              ;+0 CLOSEST SEEK DISTANCE
000B                                       ;+1 CLOSEST SEEK DIRECTION
000B              12 DTRCK   *=*+1         ;# SPACES FOR FORMAT
000C              13 DSECT   *=*+1         ;# SECTORS UNTIL DESIRED SECTOR
000D              14 CSECT   *=*+1         ;CLOSEST SEC FROM CURRENT POSITIO
000E              15 STAB    *=*+5         ;SECT HEADER TAB
0013                                       ;SAME FORMAT AS HDRS TABLE
0013              19 DRIVE   *=*+1         ;CURRENT DRIVE#
0014                  ;
0014                  ;
0014              21 TRACK   *=*+1         ;TRK# FOR CLOSEST SEEK
0015              22 NEXTS   *=*+1         ;NEXT SECTOR ON DRIVE
0016              23 SECTR   *=*+1         ;#SECT/TRK
0017              24 BUFPT   *=*+2
0019                                       ;L POINTER INTO BUFS TABLE
0019                                       ;H POINTER INTO BUFS TABLE
0019              26 HDRPT   *=*+2
001B                                       ;L POINTER INTO HDRS TABLE
001B                                       ;H POINTER INTO HDRS TABLE
001B                                       ;IF =FF THEN NO JOB
001B                  FTNUM   *=*+1         ;FORMAT TAG
001C                  IP      *=*+2
001E                  CNT     *=*+1
001F                  JOB     *=*+1         ;CURRENT JOB BEING DONE
0020                  JOBNUM  *=*+1         ;CURRENT JOB ID
0021                  DRVTRK  *=*+2         ; TRACK DRIVE IS ON CURRENTLY
0023                  STPCNT  *=*+2
0025                  CHKSUM  *=*+1
0026                  BI      *=*+1
0027                  FLG2    *=*+1
0028                  NXTJOB  *=*+1         ; OPT TRACK SEEK
0029                  NXTRK   *=*+1         ; OPT TRACK SEEK
002A                  ;
```

```
LINE# LOC   CODE          LINE

      002A                ;4K OF COMMUNITY PROPERTY
      002A                ;
      002A                      * =$400
      0400                TICK    *=*+1           ;INTERRUPT INTERVAL
      0401                DELAY   *=*+1           ;MOTOR ACCERATION DELAY  Led
      0402                CUTMT   *=*+1           ;MOTOR CUTOFF TIME       motor-off
      0403                JOBS    *=*+15
      0412                                       ;BIT 7 0:IGNORE 1:JOB PRESENT
      0412                                       ;BITS 6-4 MODE
      0412                                       ;  000:READ     (8) (0)
      0412                                       ;  001:WRITE    (9) (1)
      0412                                       ;  010:VERIFY   (A) (2)
      0412                                       ;  011:SEEK     (B) (3)
      0412                                       ;  100:BUMP     (C) (4)
      0412                                       ;  101:JUMP     (D) (5)
      0412                                       ;  110:EXECUTE  (E) (6)
      0412                                       ;  011:SECSEK   (B) (3) X011 100D
      0412                                       ;BIT 0:DRIVE 0:#B 1:#A
      0412                ;
      0412                ;
      0412                ;
      0412                      *=$421            ;HEADERS ARE AT $421
      0421                HDRS    *=*+120         ;15*8
      0499                                       ;HEADERS FOR CURRENT BLOCKS
      0499                                       ;-3:  SYNC
      0499                                       ;-2:  SYNC
      0499                                       ;-1:  "08"
      0499                                       ; 0:  ID1
      0499                                       ; 1:  ID2
      0499                                       ; 2:  TRACK# (BITS 7 & 6 PART OF ID
      0499                                       ; 3:  SECTOR#
      0499                                       ; 4:  CHECKSUM
      0499                                       ; 5:  OFF
      0499                                       ; 6,7:  SPARE
      0499                ;
      0499                TAB1    *=*+4           ;#SECTORS/TRK INITILIZED BY DOS F
      049D                GAP1    *=*+1           ;GAP 1 SIZE SET BY DOS
      049E                GAP2    *=*+1           ;USED IN FORMAT FOR MIN NUMBER OF
      049F                VERNUM  *=*+1           ; USED BY DOS
      04A0                ;
      04A0                ACTJOB  *=*+1           ;ACTIVE JOB #
      04A1                PHASE   *=*+2           ; PHASE OF STEPPER
      04A3                STPTRK  *=*+1           ; NUMBER OF STEPS PER TRACK
      04A4                NZONES  *=*+1           ; NUMBER OF ZONES
      04A5                SYNDLY  *=*+1           ; DELAY ON PLLSYN CONTROL AFTER S
      04A6                WPSW    *=*+2           ; WRITE PROTECT SWITCH
      04A8                LWPT    *=*+2
      04AA                PBI     *=*+1
      04AB                CFLG2   *=*+1
      04AC                NSIDES  *=*+1
      04AD                SPDVAR  *=*+1           ; SPEED MEASURE
      04AE                UNUSED  *=*+1           ; UNUSED
      04AF                        *=$4B0
      04B0                TAB3    *=*+8           ; TRACK BOUNDRY TABLE, UP TO 4 ZO
      04B8                ;
```

```
LINE# LOC    CODE        LINE

0171  04B8               ;
0172  04B8                                     ;DATA ON DISK PRECEEDED BY
0173  04B8                                     ;SYNC,SYNC,"07"
0174  04B8                                     ;CHKSUM FOLLOWS 256 DATA BYTES
0175  04B8                                     ;THEN APPROXIMATELY 16 SPACING BYT
0176  04B8                     *=$04EF
0177  04EF               SYNCS   *=*+1
0178  04F0               ;
0179  04F0               ;  CONSTANTS
0180  04F0               JUMP    = $D0          ;JUMP TO BUFFER CODE
0181  04F0               EXEC    = $60          ;START MOTOR THEN JUMP
0182  04F0               BUMPC   = $40          ;  RESTORE
0183  04F0               WRTSYN  = $DE          ;WRITE SYNC MODE
0184  04F0               WRTDAT  = $DC          ;WRITE DATA MODE
0185  04F0               PLLBIT  = $40          ;PLL CONTROL BIT, 1= READ MODE...
0186  04F0               FLG     = $133B
0187  04F0               SKIP2   =$2C           ;  BIT INSTRUCTION TO SKIP NEXT 2
0188  04F0               INIT    = $500
0189  04F0                       *  =$500
0190  0500               BUFS    *=*+$F00       ;=15*256
0191  1400                                     ;SET OF 15 1 BLOCK (256 WORD) BUFF
0192  1400                       .END
0193  1400                       .LIB MAINM
```

```
LINE# LOC   CODE        LINE

0175  1400                    *=$FC00        ; ROM START
0176  FC00  14 FC       STARTI .WORD START   ; INDIRECT TO START ROUTINE
0177  FC02  A1 FE       DONEI  .WORD ERROR   ; INDIRECT TO DONE ROUTINE
```

```
LINE# LOC   CODE         LINE

0199  FC04               RESET
0200  FC04  A0 02                 LDY #2         ; WAIT FOR 2 SIGNALS
0201  FC06               RESET1
0202  FC06  8C 00 04              STY $0400      ; STORE NON-ZERO FLAG.
0203  FC09               RESET2
0204  FC09  AD 00 04              LDA $0400
0205  FC0C  D0 FB                 BNE RESET2     ; WAIT FOR DOS
0206  FC0E  88                    DEY
0207  FC0F  D0 F5                 BNE RESET1     ; ONE MORE TIME
0208  FC11  4C 00 05              JMP INIT       ; DIAGNOSTICS & INIT CODE
```

```
LINE# LOC   CODE        LINE

0210  FC14              START
0211  FC14  A0 0E                 LDY #15-1       ; # OF JOBS
0212  FC16              START1
0213  FC16  58                    CLI
0214  FC17  B9 03 04              LDA JOBS,Y      ; VALID JOB?
0215  FC1A  10 32                 BPL START5      ; NO
0216  FC1C  C9 D0                 CMP #JUMP       ; JUMP CODE?
0217  FC1E  D0 04                 BNE START2      ; NO
0218  FC20  98                    TYA
0219  FC21  4C C8 FC              JMP EX2         ; GOTO EXECUTE
0220  FC24              START2
0221  FC24  78                    SEI
0222  FC25  29 01                 AND #1          ; MASK DRIVE NUMBER
0223  FC27  AA                    TAX             ; TRANSFER FOR INDEX
0224  FC28  85 13                 STA DRIVE       ; SAVE DRIVE NUMBER
0225  FC2A              ;
0226  FC2A              ; TEST MOTOR STATUS
0227  FC2A              ;
0228  FC2A  A5 40                 LDA VB
0229  FC2C  3D D5 FF   $10,$10    AND ANDC,X      ; LOOK AT MOTOR ON
0230  FC2F  F0 11                 BEQ START3      ; MOTOR IS ON
0231  FC31  45 40                 EOR VB
0232  FC33  85 40                 STA VB          ; TURN MOTOR ON
0233  FC35  A5 01                 LDA MTRCLK      ; SET TIME FOR
0234  FC37  6D 01 04              ADC DELAY       ;  ACCELERATION DELAY
0235  FC3A  95 02                 STA MTRTM,X
0236  FC3C  B5 04                 LDA DRVST,X
0237  FC3E  09 80                 ORA #$80
0238  FC40  95 04                 STA DRVST,X     ; SET ACCELERATION BIT
0239  FC42              ;
0240  FC42              ; TEST MOTOR SPEED
0241  FC42              ;
0242  FC42              START3
0243  FC42  B5 04                 LDA DRVST,X     ; FULL SPEED?
0244  FC44  30 05                 BMI START4      ; NO, STILL SET
0245  FC46  20 CA FF              JSR MTRCNT      ; YES, KEEP MOTOR ON
0246  FC49  B5 04                 LDA DRVST,X
0247  FC4B              ;
0248  FC4B              ; TEST HEAD STATUS
0249  FC4B              ;
0250  FC4B              START4
0251  FC4B  0A                    ASL A           ; HEAD MOVING?
0252  FC4C  10 05                 BPL QUE         ; NO, DO JOB
0253  FC4E              START5
0254  FC4E  88                    DEY             ; SCAN NEXT JOB
0255  FC4F  10 C5                 BPL START1
0256  FC51  30 C1                 BMI START       ; START OVER
```

```
LINE# LOC   CODE        LINE

0258  FC53              QUE
0259  FC53  58                    CLI
0260  FC54  A0 0E                 LDY #15-1        ; # OF JOBS
0261  FC56  84 20                 STY JOBNUM
0262  FC58              QUE1
0263  FC58  20 E7 FC              JSR SETJOB
0264  FC5B  30 20                 BMI QUE4
0265  FC5D              QUE2
0266  FC5D  C6 20                 DEC JOBNUM
0267  FC5F  10 F7                 BPL QUE1
0268  FC61  A4 28                 LDY NXTJOB
0269  FC63  20 E9 FC              JSR SETJB1
0270  FC66  A5 29                 LDA NXTRK        ; GET DISTANCE TO DESIRED TRACK
0271  FC68  78                    SEI
0272  FC69  95 06                 STA STEPS,X      ; STORE STEPS TO NEW TRACK
0273  FC6B  AD A3 04              LDA STPTRK       ; SET STEPS PER TRACK
0274  FC6E  95 23                 STA STPCNT,X
0275  FC70  A9 40                 LDA #$40
0276  FC72  15 04                 ORA DRVST,X
0277  FC74  95 04                 STA DRVST,X      ; SET STEPPING BIT
0278  FC76  B1 19                 LDA (HDRPT),Y
0279  FC78  95 21                 STA DRVTRK,X     ; STORE HEAD POSITION
0280  FC7A              QUE3
0281  FC7A  4C 14 FC              JMP START        ; BACK TO START
0282  FC7D              QUE4
0283  FC7D  29 01                 AND #1
0284  FC7F  C5 13                 CMP DRIVE        ; TEST IF SAME DRIVE
0285  FC81  D0 DA                 BNE QUE2         ; NOT SAME DRIVE
0286  FC83              ;
0287  FC83              ; TEST IF ON RIGHT TRACK
0288  FC83              ;
0289  FC83              QUE5
0290  FC83  B5 21                 LDA DRVTRK,X
0291  FC85  F0 0D                 BEQ GOTU
0292  FC87  38                    SEC
0293  FC88  F1 19                 SBC (HDRPT),Y
0294  FC8A  F0 08                 BEQ GOTU         ; ON RIGHT TRACK
0295  FC8C              QUE6
0296  FC8C  85 29                 STA NXTRK        ; SAVE FOR LATER USE
0297  FC8E  A5 20                 LDA JOBNUM
0298  FC90  85 28                 STA NXTJOB
0299  FC92  10 C9                 BPL QUE2
```

```
LINE# LOC   CODE        LINE

0301  FC94              GOTU
0302  FC94  A2 04                 LDX #4
0303  FC96  B1 19                 LDA (HDRPT),Y  ; LOAD TRACK NUMBER
0304  FC98  85 14                 STA TRACK
0305  FC9A              GOTU1
0306  FC9A  DD AF 04              CMP TAB3-1,X
0307  FC9D  CA                    DEX
0308  FC9E  B0 FA                 BCS GOTU1
0309  FCA0              GOTU2
0310  FCA0  BD 99 04              LDA TAB1,X     ; GET NUMBER OF SECTORS
0311  FCA3  85 16                 STA SECTR      ;  ON TRACK
0312  FCA5  8A                    TXA
0313  FCA6  0A                    ASL A
0314  FCA7  85 09                 STA WORK       ; TEMP SPACE
0315  FCA9  A5 82                 LDA PB
0316  FCAB  29 F8                 AND #$F8
0317  FCAD  05 09                 ORA WORK
0318  FCAF  05 13                 ORA DRIVE      ; ADD DRIVE
0319  FCB1  85 82                 STA PB         ; STORE BIT DENSITY
0320  FCB3  A6 13                 LDX DRIVE
0321  FCB5  B5 04                 LDA DRVST,X
0322  FCB7  30 C1                 BMI QUE3       ; MOTOR STILL ACCELERATING
0323  FCB9  A5 1F                 LDA JOB
0324  FCBB  C9 40                 CMP #BUMPC
0325  FCBD  F0 11                 BEQ BUMP       ; RESTORE HEAD TO TRACK 1
```

```
LINE# LOC   CODE        LINE

0327  FCBF              ;
0328  FCBF              ; MODE 110 (E)
0329  FCBF              ; EXECUTE
0330  FCBF              ;
0331  FCBF              EXE
0332  FCBF  C9 60              CMP #EXEC
0333  FCC1  F0 03              BEQ EX
0334  FCC3  4C 27 FE           JMP SEEK
0335  FCC6              EX
0336  FCC6  A5 20              LDA JOBNUM
0337  FCC8              EX2
0338  FCC8  18                 CLC
0339  FCC9  69 05              ADC #>BUFS
0340  FCCB  85 18              STA BUFPT+1
0341  FCCD              EX3
0342  FCCD  6C 17 00           JMP (BUFPT)     ; GO EXECUTE
```

```
LINE# LOC   CODE          LINE

0344  FCD0                ;
0345  FCD0                ; MODE 100 (C)
0346  FCD0                ; BUMP (RESTORE)
0347  FCD0                ;
0348  FCD0                BUMP
0349  FCD0  78                   SEI
0350  FCD1  A9 40                LDA #$40
0351  FCD3  95 04                STA DRVST,X
0352  FCD5  A9 01                LDA #1
0353  FCD7  95 21                STA DRVTRK,X     ; SET TRACK TO 1
0354  FCD9  BD D7 FF             LDA ANDD,X
0355  FCDC  05 40                ORA VB
0356  FCDE  85 40                STA VB           ; SET HEAD TO PHASE A
0357  FCE0  A9 5A                LDA #90
0358  FCE2  95 06                STA STEPS,X      ; DO 90 STEPS
0359  FCE4  4C 6C FE             JMP DONE         ; FINISH UP
```

```
LINE# LOC   CODE        LINE

0361  FCE7              SETJOB
0362  FCE7  A4 20              LDY JOBNUM
0363  FCE9              SETJB1
0364  FCE9  B9 03 04           LDA JOBS,Y
0365  FCEC  48                 PHA
0366  FCED  29 78              AND #$78
0367  FCEF  85 1F              STA JOB
0368  FCF1  98                 TYA
0369  FCF2  0A                 ASL A
0370  FCF3  0A                 ASL A
0371  FCF4  0A                 ASL A
0372  FCF5  69 21              ADC #<HDRS
0373  FCF7  85 19              STA HDRPT
0374  FCF9  A0 02              LDY #2
0375  FCFB  68                 PLA
0376  FCFC  60                 RTS
0377  FCFD              FSNUM
0378  FCFD  A0 03              LDY #3
0379  FCFF  B1 19              LDA (HDRPT),Y
0380  FD01  85 11              STA STAB+3
0381  FD03              FSNUM1
0382  FD03  A9 7F              LDA #$7F
0383  FD05  85 0D              STA CSECT
0384  FD07  A5 11              LDA STAB+3
0385  FD09  18                 CLC
0386  FD0A  69 02              ADC #2
0387  FD0C  C5 16              CMP SECTR
0388  FD0E  90 02              BCC FSNUM2
0389  FD10  A9 00              LDA #0
0390  FD12              FSNUM2
0391  FD12  85 15              STA NEXTS
0392  FD14  A2 0E              LDX #15-1
0393  FD16  86 20              STX JOBNUM
0394  FD18  A2 FF              LDX #$FF
0395  FD1A              FSNUM3
0396  FD1A  20 E7 FC           JSR SETJOB
0397  FD1D  10 2C              BPL FSNUM5      ; NO JOB
0398  FD1F  85 09              STA WORK
0399  FD21  29 01              AND #1
0400  FD23  C5 13              CMP DRIVE
0401  FD25  D0 24              BNE FSNUM5      ; NOT THE RIGHT DRIVE
0402  FD27  B1 19              LDA (HDRPT),Y
0403  FD29  C5 14              CMP TRACK
0404  FD2B  D0 1E              BNE FSNUM5
0405  FD2D  A5 1F              LDA JOB
0406  FD2F  C9 60              CMP #EXEC
0407  FD31  F0 18              BEQ FSNUM5
0408  FD33  C8                 INY
0409  FD34  38                 SEC
0410  FD35  B1 19              LDA (HDRPT),Y
0411  FD37  E5 15              SBC NEXTS
0412  FD39  10 03              BPL FSNUM4
0413  FD3B  18                 CLC
0414  FD3C  65 16              ADC SECTR
0415  FD3E              FSNUM4
```

```
LINE# LOC    CODE         LINE

0416  FD3E   C5 0D                  CMP CSECT
0417  FD40   B0 09                  BCS FSNUM5
0418  FD42   85 0D                  STA CSECT
0419  FD44   A5 20                  LDA JOBNUM
0420  FD46   AA                     TAX
0421  FD47   69 05                  ADC #>BUFS
0422  FD49   85 18                  STA BUFPT+1
0423  FD4B                FSNUM5
0424  FD4B   C6 20                  DEC JOBNUM
0425  FD4D   10 CB                  BPL FSNUM3
0426  FD4F   8A                     TXA
0427  FD50   10 03                  BPL FSNUM6
0428  FD52   4C 14 FC               JMP START
0429  FD55                FSNUM6
0430  FD55   8E A0 04               STX ACTJOB
0431  FD58   86 20                  STX JOBNUM
0432  FD5A   20 E7 FC               JSR SETJOB
0433  FD5D                FSNUM7
0434  FD5D   A5 1F                  LDA JOB
0435  FD5F                          .END
0436  FD5F                          .LIB R/W
```

```
LINE# LOC   CODE        LINE


0438  FD5F              ; MODE= 000 (8)
0439  FD5F              ; READ
0440  FD5F              ;
0441  FD5F  C9 00       READ   CMP #0          ; SET STATUS
0442  FD61  D0 37              BNE WRITE
0443  FD63              ;
0444  FD63  20 8B FD           JSR DSTRT       ; START READING DATA
0445  FD66              ;
0446  FD66  50 FE       L100   BVC L100        ; NO TIME FOR JSR
0447  FD68  B8                 CLV            ; CLEAR BYTE READY (V)
0448  FD69              ;
0449  FD69  A5 41              LDA DIN         ; LOAD BYTE
0450  FD6B  91 17              STA (BUFPT)Y    ; STORE IN BUFFER
0451  FD6D  C8                 INY
0452  FD6E  D0 F6              BNE L100        ;LOOP FOR 256 BYTES
0453  FD70              ;
0454  FD70  50 FE       L101   BVC L101        ; WAIT FOR CKSUM
0455  FD72  A5 41              LDA DIN
0456  FD74  85 25              STA CHKSUM
0457  FD76              ;
0458  FD76  A9 10              LDA #$10       ; TEST FOR DECODE ERROR
0459  FD78  24 4D              BIT IFR
0460  FD7A  D0 0C              BNE ERR        ; DECODE FLAG IS SET
0461  FD7C              ;
0462  FD7C  20 06 FF           JSR CHKBLK
0463  FD7F  C5 25              CMP CHKSUM     ; CMP THE CHECK SUM
0464  FD81  D0 03              BNE CSIDB
0465  FD83  4C 6C FE           JMP DONE
0466  FD86              ;
0467  FD86  A9 05       CSIDB  LDA #5          ; DATA BLOCK CS ERROR
0468  FD88  4C A1 FE    ERR    JMP ERROR
0469  FD8B              ;
0470  FD8B              ;
0471  FD8B              ;START READING DATA
0472  FD8B              DSTRT
0473  FD8B              ;
0474  FD8B  20 77 FE           JSR SRCH        ;SEARCH FOR CORRECT HEADER
0475  FD8E              ;
0476  FD8E  20 C8 FE           JSR SYNC        ;WAIT FOR START OF DATA
0477  FD91              ;
0478  FD91  C9 07              CMP #7         ; START OF DATA
0479  FD93  F0 04              BEQ L102        ;YES
0480  FD95              ;
0481  FD95  A9 04              LDA #4
0482  FD97  D0 10              BNE L104        ;DISPLAY ERROR
0483  FD99              ;
0484  FD99  60          L102   RTS
0485  FD9A                     .END
0486  FD9A                     .LIB CMDS
```

```
LINE# LOC   CODE        LINE


0488  FD9A              ;
0489  FD9A              ;  MODE 001 (9)
0490  FD9A              ;  WRITE
0491  FD9A              ;
0492  FD9A  C9 10       WRITE   CMP #$10
0493  FD9C  D0 69               BNE VRFY
0494  FD9E              ;
0495  FD9E  20 06 FF            JSR CHKBLK
0496  FDA1  85 25               STA CHKSUM
0497  FDA3              ;
0498  FDA3  A5 82               LDA PB
0499  FDA5  29 08               AND #08        ; ISOLATE WRITE PROTECT BIT
0500  FDA7  F0 03               BEQ L198       ; BRANCH IF OK
0501  FDA9              ;
0502  FDA9  4C A1 FE    L104    JMP ERROR
0503  FDAC              ;
0504  FDAC  20 77 FE    L198    JSR SRCH        ; GET CORRECT BLOCK
0505  FDAF              ;
0506  FDAF  AE 9D 04            LDX GAP1       ; LOAD WAIT GAP 1
0507  FDB2  CA                  DEX
0508  FDB3  50 FE       L200    BVC L200        ;WAIT X CHARS
0509  FDB5  B8                  CLV
0510  FDB6  CA                  DEX
0511  FDB7  D0 FA               BNE L200
0512  FDB9              ;
0513  FDB9  A9 DE               LDA #WRTSYN    ; WRITE SYNC MODE
0514  FDBB  85 4C               STA PCR
0515  FDBD  A9 DC               LDA #WRTDAT
0516  FDBF              ;
0517  FDBF  A2 FF               LDX #$FF       ; LOAD FILL CODE
0518  FDC1  20 FD FE            JSR OUT2       ; WRITE 4 SYNC CHARS OUT
0519  FDC4  20 FD FE            JSR OUT2 .     ; 1 ST
0520  FDC7              ;
0521  FDC7  50 FE       L202    BVC L202        ; 4 TH
0522  FDC9  B8                  CLV
0523  FDCA              ;
0524  FDCA  85 4C               STA PCR         ; STORE NORMAL CODE MODE
0525  FDCC              ;
0526  FDCC  A9 07               LDA #7
0527  FDCE  85 80               STA DOUT        ; 2ND SYNC
0528  FDD0              ;
0529  FDD0  A0 00               LDY #0
```

```
LINE# LOC   CODE        LINE

0531  FDD2              ; WRITE BLOCK
0532  FDD2  50 FE       L203    BVC L203
0533  FDD4  B8                  CLV
0534  FDD5              ;
0535  FDD5  B1 17               LDA (BUFPT),Y
0536  FDD7  85 80               STA DOUT
0537  FDD9              ;
0538  FDD9              ;
0539  FDD9  C8                  INY
0540  FDDA  D0 F6               BNE L203          ; LOOP FOR WHOLE BLOCK
0541  FDDC              ;
0542  FDDC  A6 25               LDX CHKSUM       ; WRITE OUT CKSUM
0543  FDDE  50 FE       L204    BVC L204
0544  FDE0  B8                  CLV
0545  FDE1  86 80               STX DOUT
0546  FDE3  20 00 FF            JSR OUT           ; SEND A BYTE
0547  FDE6  A2 00               LDX #00
0548  FDE8  20 00 FF            JSR OUT           ; SEND A BYTE
0549  FDEB  A9 FC               LDA #$FC          ; NORMAL, READ MODE
0550  FDED              ;
0551  FDED  50 FE       L442    BVC L442          ; TEST IF BYTE PRESENT
0552  FDEF  B8                  CLV
0553  FDF0              ;
0554  FDF0  85 4C               STA PCR           ; CHANGE TO SEND SYNC
0555  FDF2              ;
0556  FDF2  A2 28               LDX #40
0557  FDF4  20 F7 FE    L205    JSR BYTE
0558  FDF7  CA                  DEX
0559  FDF8  D0 FA               BNE L205
0560  FDFA              ;
0561  FDFA  A4 20               LDY JOBNUM
0562  FDFC  B9 03 04            LDA JOBS,Y       ; LOAD OLD WRITE JOB
0563  FDFF  49 30               EOR #$30         ; MAKE INTO VERIFY
0564  FE01  99 03 04            STA JOBS,Y
0565  FE04  4C FD FC            JMP FSNUM        ; FIX SECTOR# FOR NEXT JOB
```

```
LINE# LOC   CODE        LINE

0567  FE07              ; MODE= 010 (A)
0568  FE07              ; VERIFY
0569  FE07              ;
0570  FE07  C9 20       VRFY   CMP #$20        ; SECTOR SEEK
0571  FE09  D0 16              BNE SECSEK      ; SECTOR SEEK
0572  FE0B  20 8B FD           JSR DSTRT       ; START READING DATA
0573  FE0E              ;
0574  FE0E  50 FE       L210   BVC L210        ;NO TIME FOR JSR
0575  FE10  B8                 CLV
0576  FE11              ;
0577  FE11  A5 41              LDA DIN          ; GET BYTE
0578  FE13  D1 17              CMP (BUFPT),Y    ; COMPARE TO CONTENTS OF BUFFE
0579  FE15  D0 06              BNE L212         ; VERIFY ERROR
0580  FE17              ;
0581  FE17  C8                 INY
0582  FE18  D0 F4              BNE L210         ; 256 BYTES
0583  FE1A  4C 70 FD           JMP L101         ; COMPARE CKSUM BYTES
0584  FE1D  A9 07       L212   LDA #7           ; VERIFY ERROR
0585  FE1F  D0 53              BNE ERR2
0586  FE21              ;
0587  FE21              ; END READING DATA
```

```
LINE# LOC   CODE        LINE

0589  FE21              ;
0590  FE21              ; MODE= 011 (B) X011 100D
0591  FE21              ; SECTOR SEEK
0592  FE21              ;
0593  FE21              SECSEK
0594  FE21  20 77 FE           JSR SRCH        ; GET CORRECT BLOCK
0595  FE24  4C 6C FE           JMP DONE
```

```
COMMANDS......PAGE 0021

LINE# LOC   CODE        LINE


0597  FE27              ; SEEK TO DETERMINE NEXT SECTOR#
0598  FE27              ;
0599  FE27              SEEK
0600  FE27              ;
0601  FE27  A2 B4                LDX #180        ; READ HEADER
0602  FE29  20 9B FE             JSR HEAD        ; GET BLOCK HEAD
0603  FE2C              ;
0604  FE2C  A0 04                LDY #5-1
0605  FE2E  50 FE       L250     BVC L250
0606  FE30  B8                   CLV
0607  FE31  A5 41                LDA DIN        ; GET BYTE
0608  FE33  99 0E 00             STA STAB,Y     ; STORE IN TABLE
0609  FE36  88                   DEY
0610  FE37  10 F5                BPL L250       ; LOOP
0611  FE39              ;
0612  FE39  A0 04                LDY #4         ; CREATE CKSUM
0613  FE3B  A9 00                LDA #0
0614  FE3D  59 0E 00    L254     EOR STAB,Y
0615  FE40  88                   DEY
0616  FE41  10 FA                BPL L254
0617  FE43              ;
0618  FE43  C9 00                CMP #0
0619  FE45  D0 28                BNE CSERR      ; BRANCH IF ERROR
0620  FE47              ;
0621  FE47  A6 13                LDX DRIVE      ;INDEX BY DRIVE#
0622  FE49  A5 10                LDA STAB+2
0623  FE4B  95 21                STA DRVTRK,X    ; UPDATE CORRECT TRACK#
0624  FE4D              ;
0625  FE4D  A5 1F                LDA JOB         ; LOAD JOB TYPE
0626  FE4F  C9 30                CMP #$30        ; TEST IF SEEK
0627  FE51  F0 0F                BEQ ESEEK       ; BRANCH IF YES
0628  FE53              ;
0629  FE53  A0 01                LDY #1
0630  FE55              ;
0631  FE55  B1 19       L252     LDA (HDRPT),Y   ; LOAD ID
0632  FE57  D9 0E 00             CMP STAB,Y      ; CHECK IF SAME AS HEADER
0633  FE5A  D0 16                BNE L253        ; BRANCH IF MISMATCH
0634  FE5C              ;
0635  FE5C  88                   DEY
0636  FE5D  10 F6                BPL L252        ; LOOP
0637  FE5F              ;
0638  FE5F  4C 03 FD             JMP FSNUM1      ; DECIDE WHICH SECTORS TO SERVICE
```

```
LINE# LOC   CODE        LINE

0640  FE62              ; MODE= 011 (B)
0641  FE62              ; SEEK
0642  FE62              ;
0643  FE62              ESEEK
0644  FE62              ;
0645  FE62  A0 04               LDY #4
0646  FE64  B9 0E 00    L251    LDA STAB,Y         ; LOAD HEADER
0647  FE67  91 19               STA (HDRPT),Y      ; STORE HEADER
0648  FE69  88                  DEY
0649  FE6A  10 F8               BPL L251           ; LOOP FOR WHOLE HEADER
0650  FE6C              ;
0651  FE6C  A9 01       DONE    LDA #1             ; ERROR CODE FOR NO ERROR
0652  FE6E  2C                  .BYTE SKIP2
0653  FE6F  A9 09       CSERR   LDA #9
0654  FE71  2C                  .BYTE SKIP2
0655  FE72  A9 0B       L253    LDA #11
0656  FE74  4C A1 FE    ERR2    JMP ERROR
0657  FE77              ;
```

```
LINE# LOC   CODE        LINE


0659  FE77              ; SEARCH FOR SPECIFIC BLOCK
0660  FE77              ;
0661  FE77  A0 03       SRCH   LDY #4-1
0662  FE79  A9 00              LDA #0
0663  FE7B              ;
0664  FE7B  51 19       L412   EOR (HDRPT),Y   ; COMPUTE CHECKSUM
0665  FE7D  88                 DEY
0666  FE7E  10 FB              BPL L412
0667  FE80              ;
0668  FE80  A0 04              LDY #4          ; OFFSET FOR CHECKSUM
0669  FE82  91 19              STA (HDRPT),Y   ; STORE CHECKSUM
0670  FE84              ;
0671  FE84  A4 20              LDY JOBNUM
0672  FE86  A2 B4              LDX #180        ; SEARCH FOR A SECTOR
0673  FE88              ;
0674  FE88  20 9B FE    L410   JSR HEAD        ; EVERY BYTE IN THE HEADER
0675  FE8B  A0 04              LDY #5-1        ; MUST BE IDENTICAL
0676  FE8D              ;
0677  FE8D  50 FE       L411   BVC L411
0678  FE8F  B8                 CLV
0679  FE90              ;
0680  FE90  A5 41              LDA DIN
0681  FE92  D1 19              CMP (HDRPT),Y   ; COMPARE TO HEADER
0682  FE94  D0 F2              BNE L410        ; LOOP IF NOT RIGHT BLOCK
0683  FE96              ;
0684  FE96  88                 DEY
0685  FE97  10 F4              BPL L411        ; LOOP FOR WHOLE HEADER
0686  FE99              ;
0687  FE99  C8                 INY             ; SET Y TO 0
0688  FE9A              ;
0689  FE9A              ;
0690  FE9A  60                 RTS
0691  FE9B              ;
0692  FE9B              ; SEARCH FOR BLOCK HEAD
0693  FE9B              ;
0694  FE9B  58          HEAD   CLI
0695  FE9C  CA                 DEX             ; X HAS MAX # OF TRIALS
0696  FE9D  D0 21              BNE L420        ; TRIALS LEFT
0697  FE9F              ;
0698  FE9F  A9 02              LDA #2          ; ERROR CODE
```

```
LINE# LOC   CODE          LINE

0700  FEA1                ; SEND ERROR CODE
0701  FEA1                ;
0702  FEA1                ERROR
0703  FEA1  A4 20                LDY JOBNUM      ; LOAD JOB#
0704  FEA3  99 03 04             STA JOBS,Y      ; SEND JOB STATUS
0705  FEA6  48                   PHA
0706  FEA7                ;
0707  FEA7  A5 01                LDA MTRCLK
0708  FEA9  A6 13                LDX DRIVE
0709  FEAB  6D 02 04             ADC CUTMT
0710  FEAE  95 02                STA MTRTM,X     ; MAKE MOTOR STAY ON LONGER
0711  FEB0                ;
0712  FEB0  20 0F FF             JSR SETSRH
0713  FEB3                ;
0714  FEB3                ;
0715  FEB3  68                   PLA             ; LOAD ERROR TYPE
0716  FEB4  4A                   LSR A           ; TEST IF OK EXIT
0717  FEB5  D0 03                BNE L421        ; BRANCH IF NOT
0718  FEB7                ;
0719  FEB7  4C FD FC             JMP FSNUM       ; LOOP FOR ALL SECTORS
0720  FEBA                ;
0721  FEBA  A2 3F         L421   LDX #$3F
0722  FEBC  9A                   TXS             ; PURGE STACK
0723  FEBD  4C 14 FC             JMP START       ; GO TO NEXT JOB
0724  FEC0                ;
0725  FEC0  20 C8 FE      L420   JSR SYNC        ; GET A BYTE
0726  FEC3  C9 08                CMP #$08        ; COMPARE TO START OF HEADER
0727  FEC5  D0 D4                BNE HEAD
0728  FEC7  60                   RTS
0729  FEC8                ;
0730  FEC8                ; ERRORS:
0731  FEC8                ;        #1: NO ERROR
0732  FEC8                ;        #2: CAN'T FIND BLOCK HEAD
0733  FEC8                ;        #3: NO SYNC CHARACTER
0734  FEC8                ;        #4: DATA BLOCK NOT PRESENT
0735  FEC8                ;        #5: CHECKSUM ERROR
0736  FEC8                ;        #7: VERIFY ERROR
0737  FEC8                ;        #8: WRITE WITH WRITE PROTECT ON
0738  FEC8                ;        #9: CHECKSUM ERROR IN SEEKED HEADER
0739  FEC8                ;       #10: DATA RAN INTO NEXT HEADER
0740  FEC8                ;       #11: DISK ID MISMATCH
0741  FEC8                ;       #16: DECODING ERROR
```

```
LINE# LOC   CODE       LINE

      FEC8             ;
      FEC8             ; HUNT FOR SYNC CHARACTER
      FEC8             ;
      FEC8             SYNC
      FEC8  A9 1E              LDA #30
      FECA  8D EF 04           STA SYNCS
      FECD  20 0F FF   SYNC1   JSR SETSRH        ; GOTO SEARCH MODE
      FED0             ;
      FED0  78                 SEI
      FED1  A9 D0              LDA #$80+80
      FED3  85 45              STA TIMER         ; 20 MS LIMIT
      FED5             ;
      FED5  A9 03              LDA #3            ; ERROR CODE
      FED7  CE EF 04           DEC SYNCS
      FEDA  30 C5              BMI ERROR        ; > 30 FALSE SYNCS
      FEDC  24 45      L430    BIT TIMER         ; TEST IF OVER TIME LIMIT
      FEDE  10 C1              BPL ERROR         ; BRANCH IF OVER
      FEE0  24 40              BIT VB            ; TEST IF SYNC PRESENT
      FEE2  30 F8              BMI L430          ; LOOP UNTIL SYNC
      FEE4             ;
      FEE4  AC A5 04           LDY SYNDLY       ; LOAD SYNCDLY
      FEE7  24 40      WATSYN BIT VB             ; TEST SYNC PRESENT
      FEE9  30 E2              BMI SYNC1        ; GONE, FIND AGAIN
      FEEB  88                 DEY
      FEEC  D0 F9              BNE WATSYN       ; 10 US PER LOOP DELAY
      FEEE             ;
      FEEE  A5 40              LDA VB           ; GOTO READ MODE
      FEF0  09 40              ORA #PLLBIT
      FEF2  85 40              STA VB
      FEF4             ;
      FEF4  24 41              BIT DIN          ; RESET VIA LATCHING DATA
      FEF6  B8                 CLV
      FEF7             ;
      FEF7             ; GET A BYTE
      FEF7             ;
      FEF7  50 FE      BYTE    BVC BYTE          ; TEST IF BYTE PRESENT
      FEF9  B8                 CLV
      FEFA             ;
      FEFA  A5 41              LDA DIN           ; LOAD BYTE
      FEFC  60                 RTS
      FEFD             ;
      FEFD             ; BYTE TO BE SENT IS IN X
      FEFD  20 00 FF   OUT2    JSR OUT           ; DO 2 OUTS
      FF00  50 FE      OUT     BVC OUT           ;DATA IN 6530 WAS LATCHED
      FF02  B8                 CLV
      FF03  86 80              STX DOUT          ;WRITE NEW BYTE INTO PA
      FF05  60                 RTS
      FF06             ;
      FF06             ;
      FF06  A9 00      CHKBLK LDA #0             ; CREATE BLOCK CHKSUM
      FF08  A8                 TAY
      FF09  51 17      CHKB10 EOR (BUFPT),Y
      FF0B  C8                 INY
      FF0C  D0 FB              BNE CHKB10
      FF0E  60                 RTS
```

```
LINE# LOC   CODE        LINE

0798  FF0F              ;
0799  FF0F              ;
0800  FF0F              ;
0801  FF0F  A5 40       SETSRH LDA VB           ; SET PLL INTO SEARCH MODE
0802  FF11  29 BF              AND #255-PLLBIT
0803  FF13  85 40              STA VB
0804  FF15  60                 RTS
0805  FF16              ;
0806  FF16              ;
0807  FF16              ;
0808  FF16                     .END
0809  FF16                     .LIB IRQM
```

```
LINE# LOC   CODE        LINE

0811  FF16              IRQ
0812  FF16  48                    PHA
0813  FF17  8A                    TXA
0814  FF18  48                    PHA
0815  FF19  AD 00 04              LDA TICK
0816  FF1C  85 8F                 STA MITAT
0817  FF1E  E6 00                 INC CLOCK
0818  FF20  A5 00                 LDA CLOCK
0819  FF22  29 0F                 AND #$0F
0820  FF24  D0 02                 BNE IRQ1
0821  FF26  E6 01                 INC MTRCLK
0822  FF28              IRQ1
0823  FF28  A2 01                 LDX #2-1
0824  FF2A  A5 82                 LDA PB
0825  FF2C  48                    PHA               ; SAVE PB
0826  FF2D  09 01                 ORA #1
0827  FF2F              IRQ2
0828  FF2F  85 82                 STA PB            ; SELECT DRIVE
0829  FF31  A5 82                 LDA PB
0830  FF33  29 08                 AND #8            ; MASK WRITE PROTECT
0831  FF35  DD A8 04              CMP LWPT,X        ; CHECK LAST WRITE PROTECT
0832  FF38  9D A8 04              STA LWPT,X
0833  FF3B  F0 05                 BEQ IRQ3          ; NO CHANGE
0834  FF3D  A9 01                 LDA #1
0835  FF3F  9D A6 04              STA WPSW,X        ; SET SWITCH
0836  FF42              ;
0837  FF42              ; SERVICE MOTOR
0838  FF42              ;
0839  FF42              IRQ3
0840  FF42  A5 01                 LDA MTRCLK
0841  FF44  D5 02                 CMP MTRTM,X
0842  FF46  D0 11                 BNE IRQ5
0843  FF48  16 04                 ASL DRVST,X
0844  FF4A  B0 07                 BCS IRQ4
0845  FF4C  BD D5 FF              LDA ANDC,X
0846  FF4F  05 40                 ORA VB
0847  FF51  85 40                 STA VB            ; TURN MOTOR OFF
0848  FF53              IRQ4
0849  FF53  20 CA FF              JSR MTRCNT        ; KEEP MOTOR RUNNING
0850  FF56  18                    CLC
0851  FF57  76 04                 ROR DRVST,X
0852  FF59              ;
0853  FF59              ; SERVICE STEPPER
0854  FF59              ;
0855  FF59              IRQ5
0856  FF59  B5 06                 LDA STEPS,X
0857  FF5B  D0 1D                 BNE IRQ7
0858  FF5D  BD A1 04              LDA PHASE,X
0859  FF60  F0 0F                 BEQ IRQ6          ; NO PHASE OFFSET
0860  FF62  95 06                 STA STEPS,X
0861  FF64  48                    PHA
0862  FF65  A9 01                 LDA #1
0863  FF67  95 23                 STA STPCNT,X      ; QUARTER TRACK
0864  FF69  A9 00                 LDA #0
0865  FF6B  9D A1 04              STA PHASE,X
```

```
LINE# LOC   CODE        LINE

0866  FF6E  68                      PLA
0867  FF6F  D0 09                   BNE IRQ7         ; DO THE STEP
0868  FF71              IRQ6
0869  FF71  B5 04                   LDA DRVST,X      ; ON TRACK
0870  FF73  29 BF                   AND #$BF
0871  FF75  95 04                   STA DRVST,X      ; CLEAR STEPPING FLAG.
0872  FF77  4C BA FF                JMP IRQ11        ; CHECK NEXT STEPPER
0873  FF7A              IRQ7
0874  FF7A  0A                      ASL A            ; 0=OUT, 1=IN
0875  FF7B  B0 17                   BCS IRQ8         ; STEP IN
0876  FF7D              ;
0877  FF7D              ; STEP OUT CODE
0878  FF7D              ;
0879  FF7D  D6 23                   DEC STPCNT,X
0880  FF7F  D0 07                   BNE IR7
0881  FF81  AD A3 04                LDA STPTRK
0882  FF84  95 23                   STA STPCNT,X
0883  FF86  D6 06                   DEC STEPS,X
0884  FF88              IR7
0885  FF88  A5 40                   LDA VB
0886  FF8A  3D D7 FF                AND ANDD,X
0887  FF8D  38                      SEC
0888  FF8E  FD D3 FF                SBC ANDB,X
0889  FF91  4C A8 FF                JMP IRQ10
0890  FF94              ;
0891  FF94              ; STEP IN CODE
0892  FF94              ;
0893  FF94              IRQ8
0894  FF94  D6 23                   DEC STPCNT,X
0895  FF96  D0 07                   BNE IR8
0896  FF98  AD A3 04                LDA STPTRK
0897  FF9B  95 23                   STA STPCNT,X
0898  FF9D  F6 06                   INC STEPS,X
0899  FF9F              IR8
0900  FF9F  A5 40                   LDA VB
0901  FFA1  3D D7 FF                AND ANDD,X
0902  FFA4  18                      CLC
0903  FFA5  7D D3 FF                ADC ANDB,X
0904  FFA8              IRQ10
0905  FFA8  3D D7 FF                AND ANDD,X
0906  FFAB  85 08                   STA COW
0907  FFAD  A5 40                   LDA VB
0908  FFAF  3D D1 FF                AND ANDA,X
0909  FFB2  05 08                   ORA COW
0910  FFB4  85 40                   STA VB           ; WRITE NEW STEPPER STATE
0911  FFB6  A5 82                   LDA PB
0912  FFB8  29 FE                   AND #$FE         ; DRIVE 0
0913  FFBA              IRQ11
0914  FFBA  CA                      DEX
0915  FFBB  30 03                   BMI IRQ12
0916  FFBD  4C 2F FF                JMP IRQ2
0917  FFC0              IRQ12
0918  FFC0  4E AB 04                LSR CFLG2
0919  FFC3  68                      PLA
0920  FFC4  85 82                   STA PB           ; RESTORE DRIVE
```

```
LINE# LOC   CODE        LINE

0921  FFC6  68                PLA
0922  FFC7  AA                TAX
0923  FFC8  68                PLA
0924  FFC9  40                RTI
0925  FFCA              MTRCNT
0926  FFCA  A5 01             LDA MTRCLK
0927  FFCC  95 02             STA MTRTM,X
0928  FFCE  D6 02             DEC MTRTM,X
0929  FFD0  60                RTS
0930  FFD1              ;
0931  FFD1              ;
0932  FFD1              ;
0933  FFD1  F3          ANDA  .BYTE $F3,$FC
0933  FFD2  FC
0934  FFD3  04          ANDB  .BYTE $04,$01
0934  FFD4  01
0935  FFD5  20          ANDC  .BYTE $20,$10
0935  FFD6  10
0936  FFD7  0C          ANDD  .BYTE $0C,$03
0936  FFD8  03
0937  FFD9                    .END
0938  FFD9                    .LIB VECTORS
```

```
LINE# LOC   CODE       LINE

0940  FFD9             ;
0941  FFD9             ; RESET AND INTERRUPT VECTORS
0942  FFD9             ;
0943  FFD9                    *=$FFFC
0944  FFFC  04 FC             .WORD RESET
0945  FFFE  16 FF             .WORD IRQ
```

```
END......PAGE 0031

LINE# LOC   CODE        LINE


0947  0000                  .END
0948  0000                  .END


ERRORS = 0000


SYMBOL TABLE

SYMBOL VALUE
 ACR      004B     ACTJOB   04A0     ANDA     FFD1     ANDB     FFD3
 ANDC     FFD5     ANDD     FFD7     BI       0026     BUFPT    0017
 BUFS     0500     BUMP     FCD0     BUMPC    0040     BYTE     FEF7
 CFLG2    04AB     CHKB10   FF09     CHKBLK   FF06     CHKSUM   0025
 CLOCK    0000     CNT      001E     COW      0008     CSECT    000D
 CSERR    FE6F     CSIDB    FD86     CUTMT    0402     DDRB     0083
 DELAY    0401     DIN      0041     DONE     FE6C     DONEI    FC02
 DOUT     0080     DRIVE    0013     DRVST    0004     DRVTRK   0021
 DSECT    000C     DSTRT    FD8B     DTRCK    000B     EOUT     0081
 ERR      FD88     ERR2     FE74     ERROR    FEA1     ESEEK    FE62
 EX       FCC6     EX2      FCC8     EX3      FCCD     EXE      FCBF
 EXEC     0060     FLG      133B     FLG2     0027     FSNUM    FCFD
 FSNUM1   FD03     FSNUM2   FD12     FSNUM3   FD1A     FSNUM4   FD3E
 FSNUM5   FD4B     FSNUM6   FD55     FSNUM7   FD5D     FTNUM    001B
 GAP1     049D     GAP2     049E     GOTU     FC94     GOTU1    FC9A
 GOTU2    FCA0     HDRPT    0019     HDRS     0421     HEAD     FE9B
 IER      004E     IFR      004D     INIT     0500     IP       001C
 IR7      FF88     IR8      FF9F     IRQ      FF16     IRQ1     FF28
 IRQ10    FFA8     IRQ11    FFBA     IRQ12    FFC0     IRQ2     FF2F
 IRQ3     FF42     IRQ4     FF53     IRQ5     FF59     IRQ6     FF71
 IRQ7     FF7A     IRQ8     FF94     JOB      001F     JOBNUM   0020
 JOBS     0403     JUMP     00D0     L100     FD66     L101     FD70
 L102     FD99     L104     FDA9     L198     FDAC     L200     FDB3
 L202     FDC7     L203     FDD2     L204     FDDE     L205     FDF4
 L210     FE0E     L212     FE1D     L250     FE2E     L251     FE64
 L252     FE55     L253     FE72     L254     FE3D     L410     FE88
 L411     FE8D     L412     FE7B     L420     FEC0     L421     FEBA
 L430     FEDC     L442     FDED     LWPT     04A8     MITA     0080
 MITAT    008F     MTRCLK   0001     MTRCNT   FFCA     MTRTM    0002
 NEXTS    0015     NSIDES   04AC     NXTJOB   0028     NXTRK    0029
 NZONES   04A4     OUT      FF00     OUT2     FEFD     PB       0082
 PBI      04AA     PCR      004C     PHASE    04A1     PLLBIT   0040
 QUE      FC53     QUE1     FC58     QUE2     FC5D     QUE3     FC7A
 QUE4     FC7D     QUE5     FC83     QUE6     FC8C     READ     FD5F
 RESET    FC04     RESET1   FC06     RESET2   FC09     SECSEK   FE21
 SECTR    0016     SEEK     FE27     SETJB1   FCE9     SETJOB   FCE7
 SETSRH   FF0F     SKIP2    002C     SPDVAR   04AD     SRCH     FE77
 STAB     000E     START    FC14     START1   FC16     START2   FC24
 START3   FC42     START4   FC4B     START5   FC4E     STARTI   FC00
 STEPS    0006     STPCNT   0023     STPTRK   04A3     SYNC     FEC8
 SYNC1    FECD     SYNCS    04EF     SYNDLY   04A5     T1H      0045
 T1L      0044     T1LL     0044     TAB1     0499     TAB3     04B0
 TICK     0400     TIMER    0045     TRACK    0014     UNUSED   04AE
 VB       0040     VDDRB    0042     VERNUM   049F     VIAA     0040
```

```
SYMBOL TABLE

SYMBOL VALUE
 VRFY      FE07    WATSYN   FEE7    WORK     0009    WPSW     04A6
 WRITE     FD9A    WRTDAT   00DC    WRTSYN   00DE
END OF ASSEMBLY
```

```
LINE# LOC    CODE          LINE

0001  0000                        .LIB DOS


0003  0000                 ;**************************************;
0004  0000                 ;*                                    *;
0005  0000                 ;*          8050 DUAL FLOPPY          *;
0006  0000                 ;*        FILE CONTOLLER  (6502)      *;
0007  0000                 ;*        DISK OPERATING SYSTEM       *;
0008  0000                 ;*             VERSION  2.5           *;
0009  0000                 ;*        RELEASED  APRIL 05, 1980    *;
0010  0000                 ;*    MOS 2364 ROMS #901482           *;
0011  0000                 ;*                     -01,-02        *;
0012  0000                 ;*                                    *;
0013  0000                 ;*        RE-RELEASED XXXX XX, 1980   *;
0014  0000                 ;*    MOS 2364 ROMS #901482           *;
0015  0000                 ;*                     -03,-04        *;
0016  0000                 ;*                                    *;
0017  0000                 ;*   COPYRIGHT 1980                   *;
0018  0000                 ;*         COMMODORE BUSINESS MACHINES *;
0019  0000                 ;*                                    *;
0020  0000                 ;*                                    *;
0021  0000                 ;**************************************;
```

```
LINE# LOC    CODE          LINE


0023  0000                 ;   1.  DOS
0024  0000                 ;   2.  EQUATE
0025  0000                 ;   3.  I/ODEF
0026  0000                 ;   4.  RAMVAR
0027  0000                 ;   5.  ROM
0028  0000                 ;   6.  PARSEX
0029  0000                 ;   7.  SETDRV
0030  0000                 ;   8.  AUTOIT
0031  0000                 ;   9.  LOOKUP
0032  0000                 ;  10.  TRNSFR
0033  0000                 ;  11.  SCRTCH
0034  0000                 ;  12.  DUPLCT
0035  0000                 ;  13.  COPYSETUP
0036  0000                 ;  14.  COPYALL
0037  0000                 ;  15.  RENAME
0038  0000                 ;  16.  MEMRW
0039  0000                 ;  17.  BLOCK
0040  0000                 ;  18.  FNDREL
0041  0000                 ;  19.  TST2
0042  0000                 ;  20.  TST3
0043  0000                 ;  21.  TST4
0044  0000                 ;  22.  JOBS
0045  0000                 ;  23.  ADDFIL
0046  0000                 ;  24.  EROM
0047  0000                 ;  25.  OPEN
0048  0000                 ;  26.  CLOSE
0049  0000                 ;  27.  OPCHNL
0050  0000                 ;  28.  TSTFLG
0051  0000                 ;  29.  TSUTIL
0052  0000                 ;  30.  SSUTIL
0053  0000                 ;  31.  SSTEST
0054  0000                 ;  32.  GETACT
0055  0000                 ;  33.  REL1
0056  0000                 ;  34.  REL2
0057  0000                 ;  35.  REL3
0058  0000                 ;  36.  REL4
0059  0000                 ;  37.  SSEND
0060  0000                 ;  38.  RECORD
0061  0000                 ;  39.  NULBUF
0062  0000                 ;  40.  ADDREL
0063  0000                 ;  41.  NEWSS
0064  0000                 ;  42.  ERPROC
0065  0000                 ;  43.  DSKINT
0066  0000                 ;  44.  IDLE
0067  0000                 ;  45.  IEEE
0068  0000                 ;  46.  LSTDIR
0069  0000                 ;  47.  VERDIR
0070  0000                 ;  48.  NEW
0071  0000                 ;  49.  MAP
0072  0000                 ;  50.  FRETS
0073  0000                 ;  51.  TSTFND
0074  0000                 ;  52.  ROMTBL
0075  0000                 ;  53.  VECTOR
0076  0000                 ;  54.  CHKSUM
```

```
......PAGE 0003

LINE# LOC    CODE           LINE


0078  0000                         .END
0079  0000                         .LIB EQUATE
```

```
LINE# LOC   CODE          LINE


0081  0000                ;*******************************
0082  0000                ;*  EQUATES
0083  0000                ;*******************************
0084  0000                ;
0085  0000                ROM     =$C000
0086  0000                ;
0087  0000                LRF     =$80
0088  0000                DYFILE  =$40
0089  0000                OVRFLO  =$20
0090  0000                NSSL    =6
0091  0000                SSIOFF  =4+NSSL+NSSL
0092  0000                NSSP    =120
0093  0000                LOTRK   =4
0094  0000                HITRK   =LOTRK+1
0095  0000                MXCHNS  =8
0096  0000                ID      =$1000
0097  0000                MAXSA   =18
0098  0000                VERERR  =7
0099  0000                CR      =$0D              ;  CARRIAGE RETURN
0100  0000                BFCNT   =12
0101  0000                BAMJOB= BFCNT
0102  0000                BAM0    =$4100
0103  0000                BAM1    =$4200
0104  0000                CBPTR   =BAMJOB+BAMJOB+4
0105  0000                ERRCHN  =MXCHNS-1
0106  0000                ERRSA   =16
0107  0000                CMDCHN  =MXCHNS-2
0108  0000                LXINT   =$3F
0109  0000                CMDSA   =15
0110  0000                APMODE  =2
0111  0000                MDMODE  =3
0112  0000                RDMODE  =0
0113  0000                WTMODE  =1
0114  0000                RELTYP  =4
0115  0000                DIRTYP  =7
0116  0000                SEQTYP  =1
0117  0000                PRGTYP  =2
0118  0000                USRTYP  =3
0119  0000                TYPMSK  =7
0120  0000                IRSA    =17
0121  0000                IWSA    =18
0122  0000                ID8050  =$7
0123  0000                ID2040  =$F
0124  0000                ID2030  =$64
0125  0000                DOSVER  =2                ;DOS VERSION
0126  0000                FM8050  =$43              ;8050 FORMAT VERSION
0127  0000                FM2040  =$41              ;2040 FORMAT VERSION
0128  0000                FM2030  =$42              ;2030 FORMAT VERSION

0130  0000                                         ;CONTROLLER JOB TYPES
0131  0000                READ    =$80
0132  0000                WRITE   =$90
0133  0000                WVERFY  =$A0
0134  0000                SEEK    =$B0
0135  0000                SECSEK  =SEEK+8
```

```
LINE# LOC   CODE        LINE

0136  0000              BUMP   =$C0
0137  0000              JUMPC  =$D0
0138  0000              EXEC   =$E0

0140  0000              MXFILS =5                 ; MAX # FILENAMES IN STRING
0141  0000              CMDIND =30                ; CMD BUFFER INDEX (*2)
0142  0000              DIRLEN =24                ;DIRECTORY LENGTH USED
0143  0000              NBSIZ  =27                ;NAMBUF TEXT SIZE
0144  0000              CMDLEN =58

0145  0000
0146  0000                     .LIB I/ODEF
```

```
LINE# LOC   CODE         LINE


0148  0000               ;*****************************
0149  0000               ;*       I/O DEFINITIONS     *
0150  0000               ;*****************************
0151  0000               ;
0152  0000               UNLSN  =$3F              ; IEEE UNLISTEN COMMAND
0153  0000               NOTRDY =$0               ; NOT READY
0154  0000               TALKER =$80              ; IEEE TALKER FLAG
0155  0000               LISNER =1                ; IEEE LISTENER FLAG
0156  0000               EOIOUT =$80              ; TALK WITH EOI
0157  0000               EOISND =$08              ; NOT(EOI) TO SEND
0158  0000               RDYTLK =$88              ; TALK NO EOI
0159  0000               RDYLST =$1               ; READY TO LISTEN
0160  0000               RNDRDY =RDYTLK+RDYLST    ; RANDOM CHNRDY
0161  0000               RNDEOI =EOIOUT+RDYLST    ; RANDOM W/ EOI


0163  0000               ;I/O REGISTERS

0165  0000               ; MOS 6532-1

0167  0000                      * =$200
0168  0200               IEEEDI *=*+1             ;IEEE DATA IN
0169  0201               PADD1  *=*+1             ;IEEE DATA DIR
0170  0202               IEEEDO *=*+1             ;IEEE DATA OUT
0171  0203               PBDD1  *=*+1             ;IEEE DATA OUT DIR

0173  0204               ; MOS 6532-2

0175  0204                      * =$280          ;IEEE CONTROL PORT
0176  0280               PAD2   *=*+1
0177  0281               ATNA   =1
0178  0281               DACO   =2                ;(INVERTED)
0179  0281               RFDO   =4
0180  0281               EOIO   =8
0181  0281               DAVO   =16
0182  0281               EOII   =32
0183  0281               DAVI   =64
0184  0281               ATNI   =128              ;(INVERTED)
0185  0281               PADD2  *=*+1
0186  0282               PBD2   *=*+1
0187  0283                                       ;BITS 0-2 =DEVICE # SELECT
0188  0283               LED1   =$8               ;ACTIVE LED 1
0189  0283               LED0   =$10              ;ACTIVE LED 0
0190  0283               ERRLED =32               ;HARDWARE INIT ERR LED
0191  0283               NDACI  =64
0192  0283               NRFDI  =128
0193  0283               PBDD2  *=*+1
0194  0284               ATNND  *=*+1             ;ATN IS IRQ CAUSING
0195  0285               ATNPD  *=*+1
0196  0286               ATNNE  *=*+1
0197  0287               ATNPE  *=*+1
0198  0288                      .END
0199  0288                      .LIB RAMVAR
```

```
LINE# LOC   CODE        LINE

0201  0288              ; COMMON AREA DEFINES

0203  0288                      *=$1003
0204  1003              JOBS    *=*+15          ; JOB QUE
0205  1012              ;************TRKS *=*+15 ; JOB TRACK TABLE
0206  1012                      *=$1021         ; HEADERS AT $1021
0207  1021              HDRS    *=*+120         ; JOB HEADER
0208  1099                      *=$1099         ;SECTORS/TRACK TABLE
0209  1099              NUMSEC
0210  1099                      *=$109F
0211  109F              VERNUM *=*+1            ;VERSION NUMBER
0212  10A0              ACTJOB *=*+1            ;CONTROLLER'S ACTIVE JOB
0213  10A1              PHASE   *=*+2           ;STEPPER BASE PHASE
0214  10A3              STPTRK *=*+1            ;NUMBER OF TRACKS/STEP
0215  10A4              NZONES *=*+1            ;NUMBER OF DENSITY ZONES
0216  10A5              SYNDLY *=*+1            ;SYNC DELAY FOR PLL
0217  10A6              WPSW    *=*+2           ;WRITE PROTECT CHANGE FLAG
0218  10A8              LWPT    *=*+2           ;LAST STATE OF WP SWITCH
0219  10AA              PBI     *=*+1           ;BLOCK IDENTIFIER
0220  10AB              CFLG2   *=*+1           ;COMMON FLAG2
0221  10AC              NSIDES *=*+1            ;# OF SIDES ON DISKETTE
0222  10AD              ;******* EXPAND COMMON VARIABLES HERE *****
0223  10AD              ;
0224  10AD                      *=$10B0         ;START OF TRACK BOUNDARIES
0225  10B0              MAXTRK                  ;MAX TRACK # +1
0226  10B0              TRKNUM *=*+8            ;# OF 1ST TRACK IN EACH...
0227  10B8              OFFSET *=*+8            ;RECOVERY TRACK OFFSET SEQ.
0228  10C0                                     ;...ZONE, BUT 1ST ZONE.  (REVER
0229  10C0              ;PERMANENT ADDRESS VARIABLES
0230  10C0              ;
0231  10C0                      *=$10F0
0232  10F0              VNMI    *=*+2           ;INDIRECT FOR NMI
0233  10F2              NMIFLG *=*+1
0234  10F3              AUTOFG *=*+1
0235  10F4              SECINC *=*+1            ;SECTOR INC FOR SEQ
0236  10F5              REVCNT *=*+1            ; ERROR RECOVERY COUNT

0238  10F6              BUFS    = $1100         ; START OF DATA BUFS
0239  10F6              FBUFS   = $1D00         ;FORMAT DOWNLOAD IMAGE


0241  10F6              ;*
0242  10F6              ;********************************
0243  10F6              ;*
0244  10F6              ;*      ZERO PAGE VARIABLES
0245  10F6              ;*
0246  10F6              ;********************************
0247  10F6              ;*
0248  10F6                      *= $0
0249  0000              USRJMP *=*+2            ; USER JMP TABLE PTR
0250  0002              BMPNT   *=*+2           ; BIT MAP POINTER
0251  0004              TEMP    *=*+6           ; TEMP WORK SPACE
0252  000A              IP      *=*+2           ; INDIRECT PTR VARIABLE
0253  000C              LSNADR *=*+1            ; LISTEN ADDRESS
0254  000D              TLKADR *=*+1            ;TALKER ADDRESS
```

```
LINE# LOC   CODE          LINE

0255  000E                LSNACT *=*+1            ; ACTIVE LISTENER FLAG
0256  000F                TLKACT *=*+1            ; ACTIVE TALKER FLAG
0257  0010                ADRSED *=*+1            ; ADDRESSED FLAG
0258  0011                PRGTRK *=*+1            ;LAST PROG ACCESSED
0259  0012                DRVNUM *=*+1            ;CURRENT DRIVE #
0260  0013                TRACK  *=*+1            ;CURRENT TRACK
0261  0014                SECTOR *=*+1            ;CURRENT SECTOR
0262  0015                LINDX  *=*+1            ;LOGICAL INDEX
0263  0016                SA     *=*+1            ;SECONDARY ADDRESS
0264  0017                ORGSA  *=*+1            ;ORIGINAL SA
0265  0018                DATA   *=*+1            ; TEMP DATA BYTE
0266  0019                ;*
0267  0019                ;*
0268  0019                T0     =TEMP
0269  0019                T1     =TEMP+1
0270  0019                T2     =TEMP+2
0271  0019                T3     =TEMP+3
0272  0019                T4     =TEMP+4
0273  0019                R0     *=*+1
0274  001A                R1     *=*+1
0275  001B                R2     *=*+1
0276  001C                R3     *=*+1
0277  001D                R4     *=*+1
0278  001E                RESULT *=*+4
0279  0022                ACCUM  *=*+5
0280  0027                DIRBUF *=*+2
0281  0029                ;*
0282  0029                ;*******************
0283  0029                ;*
0284  0029                ;*  ZERO PAGE ARRAY
0285  0029                ;*
0286  0029                ;*********************
0287  0029                ;*
0288  0029                BUFTAB *=*+CBPTR+4      ; BUFFER BYTE POINTERS
0289  0049                CB=BUFTAB+CBPTR
0290  0049                BUF0   *=*+MXCHNS
0291  0051                BUF1   *=*+MXCHNS
0292  0059                NBKL
0293  0059                RECL   *=*+MXCHNS
0294  0061                NBKH
0295  0061                RECH   *=*+MXCHNS
0296  0069                NR     *=*+MXCHNS
0297  0071                RS     *=*+MXCHNS
0298  0079                SS     *=*+MXCHNS
0299  0081                F1PTR  *=*+1            ; FILE STREAM 1 POINTER
0300  0082                ;
0301  0082                ;*********************
0302  0082                ; $4300 VARS MOVED TO ZP
0303  0082                ;
0304  0082                RECPTR *=*+1
0305  0083                SSNUM  *=*+1
0306  0084                SSIND  *=*+1
0307  0085                RELPTR *=*+1
0308  0086                ENTSEC *=*+MXFILS       ; SECTOR OF DIRECTORY ENTRY
0309  008B                ENTIND *=*+MXFILS       ; INDEX OF DIRECTORY ENTRY
```

```
LINE# LOC   CODE        LINE

0310  0090              FILDRV *=*+MXFILS       ; DEFAULT FLAG, DRIVE #
0311  0095              PATTYP *=*+MXFILS       ; PATTERN,REPLACE,CLOSED-FLA
0312  009A              FILTYP *=*+MXCHNS       ; CHANNEL FILE TYPE
0313  00A2              CHNRDY *=*+MXCHNS       ; CHANNEL STATUS
0314  00AA              EOIFLG *=*+1            ; TEMP EOI
0315  00AB              JOBNUM *=*+1            ; CURRENT JOB #
0316  00AC              LINTAB *=*+MAXSA+1      ; SA:LINDX TABLE
0317  00BF              CHNDAT *=*+MXCHNS       ; CHANNEL DATA BYTE
0318  00C7              LSTCHR *=*+MXCHNS       ; CHANNEL LAST CHAR PTR
0319  00CF              TYPE   *=*+1            ; ACTIVE FILE TYPE
0320  00D0              ;
0321  00D0              ;*
0322  00D0              ;******************
0323  00D0              ;*
0324  00D0              ;* RAM VARIABLES IN $4300
0325  00D0              ;*
0326  00D0              ;******************
0327  00D0              ;*
0328  00D0                     *=$4300
0329  4300              CMDBUF *=*+CMDLEN+1
0330  433B              CMDNUM *=*+1            ; COMMAND #
0331  433C              STRSIZ *=*+1
0332  433D              ;ZP:   RECPTR *=*+1
0333  433D              ;ZP:   SSNUM  *=*+1
0334  433D              ;ZP:   SSIND  *=*+1
0335  433D              ;ZP:   RELPTR *=*+1

0337  433D              TEMPSA *=*+1            ; TEMPORARY SA
0338  433E              ;ZP:   EOIFLG *=*+1          ; TEMP EOI
0339  433E              CMD    *=*+1            ; TEMP JOB COMMAND

0341  433F              LSTSEC *=*+1            ;
0342  4340              BUFUSE *=*+2            ; BUFFER ALLOCATION
0343  4342              ;ZP:   JOBNUM *=*+1          ; CURRENT JOB #
0344  4342              DSKID  *=*+4            ; CURRENT DISK ID'S
0345  4346              MDIRTY *=*+2            ;BAM 0 & 1 DIRTY FLAGS
0346  4348              ENTFND *=*+1            ;DIR-ENTRY FOUND FLAG
0347  4349              DIRLST *=*+1            ;DIR LISTING FLAG
0348  434A              CMDWAT *=*+1            ;COMMAND WAITING FLAG
0349  434B              LINUSE *=*+1            ;LINDX USE WORD
0350  434C              LBUSED *=*+1            ;LAST BUFFER USED
0351  434D              REC    *=*+1
0352  434E              TRKSS  *=*+1
0353  434F              SECSS  *=*+1
```

```
LINE# LOC   CODE         LINE

0355  4350               ;*
0356  4350               ;********************************
0357  4350               ;*
0358  4350               ;*  RAM ARRAY AREA
0359  4350               ;*
0360  4350               ;********************************
0361  4350               ;*
0362  4350               LSTJOB *=*+BFCNT+3      ; LAST JOB
0363  435F               ;ZP:   LINTAB *=*+MAXSA+1      ; SA:LINDX TABLE
0364  435F               ;ZP:   CHNDAT *=*+MXCHNS       ; CHANNEL DATA BYTE
0365  435F               DSEC   *=*+MXCHNS       ; SECTOR OF DIRECTORY ENTRY
0366  4367               DIND   *=*+MXCHNS       ; INDEX OF DIRECTORY ENTRY
0367  436F               ERWORD *=*+1            ; ERROR WORD FOR RECOVERY
0368  4370               PRGDRV *=*+1            ; LAST PROGRAM DRIVE
0369  4371               PRGSEC *=*+1            ; LAST PROGRAM SECTOR
0370  4372               WLINDX *=*+1            ; WRITE LINDX
0371  4373               RLINDX *=*+1            ; READ LINDX
0372  4374               NBTEMP *=*+2            ; # BLOCKS TEMP


0374  4376               CMDSIZ *=*+1            ; COMMAND STRING SIZE
0375  4377               CHAR   *=*+1            ; CHAR UNDER PARSER
0376  4378               LIMIT  *=*+1            ; PTR LIMIT IN COMPAR
0377  4379               F1CNT  *=*+1            ; FILE STREAM 1 COUNT
0378  437A               F2CNT  *=*+1            ; FILE STREAM 2 COUNT
0379  437B               F2PTR  *=*+1            ; FILE STREAM 2 POINTER
0380  437C               ;  PARSER TABLES
0381  437C               FILTBL *=*+MXFILS+1     ; FILENAME POINTER
0382  4382               ;ZP:   FILENT *=*+MXFILS       ; DIRECTORY ENTRY
0383  4382               ;ZP:   FILDAT *=*+MXFILS       ; DRIVE #, PATTERN
0384  4382               FILTRK *=*+MXFILS       ; 1ST LINK/TRACK
0385  4387               FILSEC *=*+MXFILS       ;         /SECTOR

0387  438C               ;  CHANNEL TABLES
0388  438C               ;ZP:   FILTYP *=*+MXCHNS ; CHANNEL FILE TYPE
0389  438C               ;ZP:   CHNRDY *=*+MXCHNS       ; CHANNEL STATUS
0390  438C               ;ZP:    LSTCHR *=*+MXCHNS       ; CHANNEL LAST CHAR PT

0392  438C               PATFLG *=*+1            ; PATTERN PRESENCE FLAG
0393  438D               IMAGE  *=*+1            ; FILE STREAM IMAGE
0394  438E               DRVCNT *=*+1            ; NUMBER OF DRV SEARCHES
0395  438F               DRVFLG *=*+1            ; DRIVE SEARCH FLAG
0396  4390               LSTDRV *=*+1            ; LAST DRIVE W/O ERROR
0397  4391               FOUND  *=*+1            ; FOUND FLAG IN DIR SEARCHES
0398  4392               DIRSEC *=*+1            ; DIRECTORY SECTOR
0399  4393               DELSEC *=*+1            ; SECTOR OF 1ST AVAIL ENTRY
0400  4394               DELIND *=*+1            ; INDEX  "
0401  4395               LSTBUF *=*+1            ; =0 IF LAST BLOCK
0402  4396               INDEX  *=*+1            ; CURRENT INDEX IN BUFFER
0403  4397               FILCNT *=*+1            ; COUNTER, FILE ENTRIES
0404  4398               TYPFLG *=*+1            ; MATCH BY TYPE FLAG
0405  4399               MODE   *=*+1            ; ACTIVE FILE MODE (R,W)
0406  439A               ;ZP:  TYPE   *=*+1            ; ACTIVE FILE TYPE
0407  439A               JOBRTN *=*+1           ;JOB RETURN FLAG
```

```
LINE# LOC    CODE          LINE


0408  439B                 EPTR    *=*+1              ;PTR FOR RECOVERY
0409  439C                 TOFF    *=*+1              ;TOTAL TRACK OFFSET
0410  439D                 NDBL    *=*+2
0411  439F                 NDBH    *=*+2
0412  43A1                 NODRV   *=*+2

0414  43A3                                           ;*
0415  43A3                 ;*****************************************
0416  43A3                 ;*
0417  43A3                 ;*   OUTPUT BUFFERS
0418  43A3                 ;*
0419  43A3                 ;*******************************************
0420  43A3                 ;*
0421  43A3                         *=$4400-36-36
0422  43B8                 NAMBUF *=*+36              ; DIRECTORY BUFFER
0423  43DC                 ERRBUF =*                  ; ERROR MSG BUFFER


0425  43DC                         .END
0426  43DC                         .LIB ROM
0427  43DC                         *=ROM
0428  C000                 CODE                       ;CONTROLLER FORMAT CODE
0429  C000                         *=*+$3A1
0430  C3A1                 CDIAG                      ;CONTROLLER POWER UP...
0431  C3A1                         *=*+$80           ;...DIAGNOSTICS & INIT.
0432  C421   00            CCHKSM .BYTE 0
0433  C422                         .END
0434  C422                         .LIB PARSEX
```

```
LINE# LOC   CODE         LINE

0436  C422               ;PARSE & EXECUTE STRING IN CMDBUF

0438  C422               PARSXQ
0439  C422  20 71 EF            JSR OKERR
0440  C425  A5 17               LDA ORGSA
0441  C427  10 09               BPL PS05
0442  C429  29 0F               AND #$F
0443  C42B  C9 0F               CMP #$F
0444  C42D  F0 03               BEQ PS05
0445  C42F  4C 01 E0            JMP OPEN
0446  C432  20 81 C5     PS05   JSR CMDSET      ;SET VARIABLES,REGS
0447  C435  B1 45               LDA (CB),Y
0448  C437  8D 77 43            STA CHAR
0449  C43A  A2 0B               LDX #NCMDS-1   ;SEARCH CMD TABLE
0450  C43C  BD 50 FB     PS10   LDA CMDTBL,X
0451  C43F  CD 77 43            CMP CHAR
0452  C442  F0 08               BEQ PS20
0453  C444  CA                  DEX
0454  C445  10 F5               BPL PS10
0455  C447  A9 31               LDA #BADCMD    ;NO SUCH CMD
0456  C449  4C 96 C4            JMP CMDERR
0457  C44C  8E 3B 43     PS20   STX CMDNUM      ;X= CMD #

0459  C44F  E0 09               CPX #PCMD      ;CMDS NOT PARSED
0460  C451  90 03               BCC PS30
0461  C453  20 BC C4            JSR TAGCMD     ;SET TABLES, POINTERS &PATTERN
0462  C456  AE 3B 43     PS30   LDX CMDNUM
0463  C459  BD 5C FB            LDA CJUMPL,X
0464  C45C  85 04               STA TEMP
0465  C45E  BD 68 FB            LDA CJUMPH,X
0466  C461  85 05               STA TEMP+1
0467  C463  6C 04 00            JMP (TEMP)     ;COMMAND TABLE JUMP

0469  C466               ; SUCCESSFUL COMMAND TERMINATION
0470  C466               ENDCMD
0471  C466  AD 6F 43            LDA ERWORD
0472  C469  D0 2B               BNE CMDERR
0473  C46B               ;
0474  C46B  A0 00               LDY #0
0475  C46D  98                  TYA
0476  C46E  84 13               STY TRACK

0478  C470  84 14        SCREND STY SECTOR      ;SCRATCH ENTRY
0479  C472  84 45               STY CB
0480  C474  20 7C EF            JSR ERRMSG
0481  C477  20 F2 F9            JSR ERROFF
0482  C47A  A5 12               LDA DRVNUM
0483  C47C  8D 90 43            STA LSTDRV
0484  C47F  AA                  TAX
0485  C480  A9 00               LDA #0
0486  C482  9D A1 43            STA NODRV,X
0487  C485  20 8B C4            JSR CLRCB
0488  C488  4C D8 D7            JMP FREICH     ;FREE INTERNAL CHANNEL
0489  C48B               ;
0490  C48B               CLRCB
```

```
LINE# LOC   CODE        LINE


0491  C48B  A0 39              LDY #CMDLEN-1
0492  C48D  A9 00              LDA #0
0493  C48F              CLRB2
0494  C48F  99 00 43           STA CMDBUF,Y
0495  C492  88                 DEY
0496  C493  10 FA              BPL CLRB2
0497  C495  60                 RTS
0498  C496              ;
0499  C496              ; COMMAND LEVEL ERROR PROCESSING
0500  C496  A0 00       CMDERR LDY #0
0501  C498  84 13              STY TRACK
0502  C49A  84 14              STY SECTOR
0503  C49C  4C EA EE           JMP CMDER2

0505  C49F  A2 00       SIMPRS LDX #0          ;SIMPLE PARSER
0506  C4A1  8E 7C 43           STX FILTBL
0507  C4A4  A9 3A              LDA #':
0508  C4A6  20 36 C5           JSR PARSE
0509  C4A9  F0 05              BEQ SP10
0510  C4AB  88                 DEY
0511  C4AC  88                 DEY
0512  C4AD  8C 7C 43           STY FILTBL
0513  C4B0  4C 33 C6    SP10   JMP SETANY      ;SET DRIVE #
```

```
LINE# LOC   CODE        LINE

0515  C4B3              PRSCLN
0516  C4B3  A0 00              LDY #0
0517  C4B5  A2 00              LDX #0
0518  C4B7  A9 3A              LDA #':
0519  C4B9  4C 36 C5           JMP PARSE       ;FIND POS'N OF ":"
0520  C4BC              ;
0521  C4BC              ;TAG COMMAND STRING
0522  C4BC              ;  SET UP CMD STRUCTURE
0523  C4BC              ;  IMAGE & FILE STREAM PTRS
0524  C4BC              ;

0526  C4BC              TAGCMD
0527  C4BC  20 B3 C4           JSR PRSCLN
0528  C4BF  D0 05              BNE TC30
0529  C4C1  A9 34       TC25   LDA #NOFILE      ;NONE, NO FILES
0530  C4C3  4C 96 C4           JMP CMDERR
0531  C4C6  88          TC30   DEY
0532  C4C7  88                 DEY
0533  C4C8  8C 7C 43           STY FILTBL      ;":"-1 STARTS FS1
0534  C4CB  8A                 TXA
0535  C4CC  D0 F3              BNE TC25        ;ERR: "," BEFORE ":"
0536  C4CE  A9 3D       TC35   LDA #'=          ;SEARCH: "="
0537  C4D0  20 36 C5           JSR PARSE
0538  C4D3  8A                 TXA             ;?FILE COUNT= 1-1?
0539  C4D4  F0 02              BEQ TC40
0540  C4D6  A9 40              LDA #%01000000 ;G1-BIT
0541  C4D8  09 21       TC40   ORA #%00100001  ;E1,^E2-BITS
0542  C4DA  8D 8D 43           STA IMAGE       ;FS STRUCTURE
0543  C4DD  E8                 INX
0544  C4DE  8E 79 43           STX F1CNT
0545  C4E1  8E 7A 43           STX F2CNT       ;INIT FOR NO FS2
0546  C4E4  AD 8C 43           LDA PATFLG
0547  C4E7  F0 0D              BEQ TC50
0548  C4E9  A9 80              LDA #%10000000 ;P1-BIT
0549  C4EB  0D 8D 43           ORA IMAGE
0550  C4EE  8D 8D 43           STA IMAGE
0551  C4F1  A9 00              LDA #0
0552  C4F3  8D 8C 43           STA PATFLG      ;CLEAR PATTERN FLAG
0553  C4F6  98          TC50   TYA              ;PTR TO FS2
0554  C4F7  F0 29              BEQ TC75        ;  FS2 NOT HERE
0555  C4F9  9D 7C 43           STA FILTBL,X
0556  C4FC  AD 79 43           LDA F1CNT       ;FS2 IS HERE NOW,...
0557  C4FF  8D 7B 43           STA F2PTR       ;...NOW SET F2 PTR
0558  C502  A9 8D              LDA #$8D        ;FIND CR-SHIFTED
0559  C504  20 36 C5           JSR PARSE       ;PARSE REST OF CMD STRING
0560  C507  E8                 INX             ;ADVANCE FILTBL PTR TO END
0561  C508  8E 7A 43           STX F2CNT       ;SAVE IT
0562  C50B  CA                 DEX             ;RESTORE FOR TEST
0563  C50C  AD 8C 43           LDA PATFLG      ;SAVE LAST PATTERN
0564  C50F  F0 02              BEQ TC60        ;?ANY PATTERNS?
0565  C511  A9 08              LDA #%1000      ;YES, P2-BIT
0566  C513  EC 79 43    TC60   CPX F1CNT        ;?F2CNT=F1CNT+1?
0567  C516  F0 02              BEQ TC70
0568  C518  09 04              ORA #%0100      ;G2-BIT
0569  C51A  09 03       TC70   ORA #%0011       ;E2-BIT,^E2-BIT
```

```
LINE# LOC   CODE        LINE

0570  C51C  4D 8D 43          EOR IMAGE        ;EOR CLEARS ^E2-BIT
0571  C51F  8D 8D 43          STA IMAGE
0572  C522              TC75
0573  C522  AD 8D 43          LDA IMAGE
0574  C525  AE 3B 43          LDX CMDNUM
0575  C528  3D 6C FB          AND STRUCT,X     ;MATCH CMD TEMPLATE
0576  C52B  D0 01             BNE TC80
0577  C52D  60                RTS
0578  C52E  8D 6F 43    TC80  STA ERWORD        ;**COULD BE WARNING
0579  C531  A9 30             LDA #BADSYN      ;ERR: BAD SYNTAX
0580  C533  4C 96 C4          JMP CMDERR
```

```
LINE# LOC   CODE        LINE


0583  C536              ;PARSE STRING
0584  C536              ;  LOOKS FOR SPECIAL CHARS,
0585  C536              ;  RETURNING WHEN VAR'BL CHAR
0586  C536              ;  IS FOUND
0587  C536              ;   A: VAR'BL CHAR
0588  C536              ;   X: IN,OUT: INDEX, FILTBL+1
0589  C536              ;   Y: IN: INDEX, CMDBUF
0590  C536              ;     OUT: NEW PTR, =0 IF NONE
0591  C536              ;         (Z=1) IF Y=0

0593  C536  8D 77 43    PARSE  STA CHAR        ;SAVE VAR'BL CHAR
0594  C539  CC 76 43    PR10   CPY CMDSIZ      ;STAY IN STRING
0595  C53C  B0 2E              BCS PR30
0596  C53E  B1 45              LDA (CB),Y     ;MATCH CHAR
0597  C540  C8                 INY
0598  C541  CD 77 43           CMP CHAR
0599  C544  F0 28              BEQ PR35       ;FOUND CHAR
0600  C546  C9 2A              CMP #'*        ;MATCH PATTERN CHARS
0601  C548  F0 04              BEQ PR20
0602  C54A  C9 3F              CMP #'?
0603  C54C  D0 03              BNE PR25
0604  C54E  EE 8C 43    PR20   INC PATFLG      ;SET PATTERN FLAG
0605  C551  C9 2C       PR25   CMP #',         ;MATCH FILE SEPARATOR
0606  C553  D0 E4              BNE PR10
0607  C555  98                 TYA
0608  C556  9D 7D 43           STA FILTBL+1,X ;PUT PTRS IN TABLE
0609  C559  AD 8C 43           LDA PATFLG     ;SAVE PATTERN FOR EA FILE
0610  C55C  29 7F              AND #$7F
0611  C55E  F0 07              BEQ PR28
0612  C560  A9 80              LDA #$80       ;RETAIN PATTERN PRESENCE...
0613  C562  95 95              STA PATTYP,X
0614  C564  8D 8C 43           STA PATFLG     ;...BUT CLEAR COUNT
0615  C567  E8          PR28   INX
0616  C568  E0 04              CPX #MXFILS-1
0617  C56A  90 CD              BCC PR10       ;NO MORE THAN MXFILS
0618  C56C  A0 00       PR30   LDY #0          ;Y=0 (Z=1)
0619  C56E  AD 76 43    PR35   LDA CMDSIZ
0620  C571  9D 7D 43           STA FILTBL+1,X
0621  C574  AD 8C 43           LDA PATFLG
0622  C577  29 7F              AND #$7F
0623  C579  F0 04              BEQ PR40
0624  C57B  A9 80              LDA #$80
0625  C57D  95 95              STA PATTYP,X
0626  C57F  98          PR40   TYA             ;Z IS SET
0627  C580  60                 RTS

0629  C581              ;INITIALIZE COMMAND TABLES, PTRS, ETC.

0631  C581  A4 45       CMDSET LDY BUFTAB+CBPTR
0632  C583  F0 14              BEQ CS08
0633  C585  88                 DEY
0634  C586  F0 10              BEQ CS07
0635  C588  B9 00 43           LDA CMDBUF,Y
0636  C58B  C9 0D              CMP #CR
0637  C58D  F0 0A              BEQ CS08
```

```
LINE# LOC   CODE          LINE


0638  C58F  88                      DEY
0639  C590  B9 00 43                LDA CMDBUF,Y
0640  C593  C9 0D                   CMP #CR
0641  C595  F0 02                   BEQ CS08
0642  C597  C8                      INY
0643  C598  C8            CS07      INY
0644  C599  8C 76 43      CS08      STY CMDSIZ       ;SET CMD STRING SIZE

0646  C59C  C0 3B                   CPY #CMDLEN+1
0647  C59E  A0 FF                   LDY #$FF
0648  C5A0  90 08                   BCC CMDRST
0649  C5A2  8C 3B 43                STY CMDNUM
0650  C5A5  A9 32                   LDA #LONGLN     ;LONG LINE ERROR
0651  C5A7  4C 96 C4                JMP CMDERR
0652  C5AA                ;  CLEAR VARIABLES,TABLES
0653  C5AA                CMDRST
0654  C5AA  A0 00                   LDY #0
0655  C5AC  98                      TYA
0656  C5AD  85 45                   STA BUFTAB+CBPTR
0657  C5AF  8D 4D 43                STA REC
0658  C5B2  85 CF                   STA TYPE
0659  C5B4  8D 98 43                STA TYPFLG
0660  C5B7  85 81                   STA F1PTR
0661  C5B9  8D 7B 43                STA F2PTR
0662  C5BC  8D 79 43                STA F1CNT
0663  C5BF  8D 7A 43                STA F2CNT
0664  C5C2  8D 8C 43                STA PATFLG
0665  C5C5  8D 6F 43                STA ERWORD
0666  C5C8  A2 05                   LDX #MXFILS
0667  C5CA  9D 7B 43      CS10      STA FILTBL-1,X
0668  C5CD  95 85                   STA ENTSEC-1,X
0669  C5CF  95 8A                   STA ENTIND-1,X
0670  C5D1  95 8F                   STA FILDRV-1,X
0671  C5D3  95 94                   STA PATTYP-1,X
0672  C5D5  9D 81 43                STA FILTRK-1,X
0673  C5D8  9D 86 43                STA FILSEC-1,X
0674  C5DB  CA                      DEX
0675  C5DC  D0 EC                   BNE CS10
0676  C5DE  60                      RTS

0678  C5DF                          .END
0679  C5DF                          .LIB SETDRV
```

```
LINE# LOC   CODE         LINE

0681  C5DF               ;SET 1ST DRIVE AND TABLE POINTERS
0682  C5DF  AD 7A 43     ONEDRV LDA F2CNT
0683  C5E2  8D 79 43            STA F1CNT
0684  C5E5  A9 01               LDA #1
0685  C5E7  8D 7A 43            STA F2CNT
0686  C5EA  8D 7B 43            STA F2PTR

0688  C5ED               ;SET UP ALL DRIVES FROM F2CNT
0689  C5ED  AC 90 43     ALLDRS LDY LSTDRV      ;SET UP DRIVE #'S...
0690  C5F0  A2 00               LDX #0         ;...INTO FILE ENTRY TABLE...
0691  C5F2  86 81        AD10   STX F1PTR       ;...ON SECTOR PTR BYTE
0692  C5F4  BD 7C 43            LDA FILTBL,X
0693  C5F7  20 09 C6            JSR SETDRV
0694  C5FA  A6 81               LDX F1PTR
0695  C5FC  9D 7C 43            STA FILTBL,X    ;INCR PTR PAST ":"
0696  C5FF  98                  TYA             ;BITS REP DRIVES
0697  C600  95 90               STA FILDRV,X    ;BIT7: DEFAULT
0698  C602  E8                  INX             ;BIT0: DRIVE #
0699  C603  EC 7A 43            CPX F2CNT
0700  C606  90 EA               BCC AD10
0701  C608  60                  RTS

0703  C609               ;SET DRIVE NUMBER
0704  C609               ;  DETERMINES DRIVE # FROM TEXT OR
0705  C609               ;  USES DEFAULT (-D)
0706  C609               ;  A: IN,OUT: INDEX, CMDBUF
0707  C609               ;  Y: IN: DEFAULT DRIVE
0708  C609               ;     OUT: DRIVE NUMBER, - IF DEFAULT

0710  C609  AA           SETDRV TAX              ;X= CMDBUF INDEX
0711  C60A  A9 3A               LDA #':
0712  C60C  DD 01 43            CMP CMDBUF+1,X  ;FOR XXX:FILE
0713  C60F  F0 0C               BEQ SD40        ;       ^
0714  C611  DD 00 43            CMP CMDBUF,X    ;FOR XXX:FILE
0715  C614  D0 16               BNE SD50        ;        ^
0716  C616  E8                  INX             ;FOUND ":", SO...
0717  C617  98           SD20   TYA              ;DRIVE= DEFAULT
0718  C618  29 01        SD22   AND #1           ;CONVERT TO NUMERIC
0719  C61A  A8           SD24   TAY              ;RESTORE DRIVE
0720  C61B  8A                  TXA             ;A=INDEX & XXXXFILE
0721  C61C  60                  RTS             ;            ^


0723  C61D  BD 00 43     SD40   LDA CMDBUF,X
0724  C620  E8                  INX             ;  XXX:FILE
0725  C621  E8                  INX             ;    --^
0726  C622  C9 30               CMP #'0         ;FOR XX0:FILE
0727  C624  F0 F2               BEQ SD22        ;          ^
0728  C626  C9 31               CMP #'1         ;FOR XX1:FILE
0729  C628  F0 EE               BEQ SD22        ;          ^
0730  C62A  D0 EB               BNE SD20        ;CMD:FILE OR XX,:FILE
0731  C62C               SD50   =*               ;   ^           ^
0732  C62C  98                  TYA             ;FOR XXX,FILE OR XX=FILE
0733  C62D  09 80               ORA #$80        ;       ^          ^
0734  C62F  29 81               AND #$81        ;DRIVE= -DEFAULT
```

```
LINE# LOC   CODE         LINE

0735  C631  D0 E7                BNE SD24        ;FINISH TESTING

0737  C633               ;SET DRIVE FROM ANY CONFIG.
0738  C633  A9 00        SETANY LDA #0
0739  C635  8D 8D 43            STA IMAGE
0740  C638  AC 7C 43            LDY FILTBL
0741  C63B  B1 45        SA05   LDA (CB),Y
0742  C63D  20 89 C6            JSR TSTOV1
0743  C640  10 12               BPL SA20
0744  C642  C8                  INY
0745  C643  CC 76 43            CPY CMDSIZ
0746  C646  B0 06               BCS SA10
0747  C648  AC 76 43            LDY CMDSIZ
0748  C64B  88                  DEY
0749  C64C  D0 ED               BNE SA05
0750  C64E  CE 8D 43     SA10   DEC IMAGE
0751  C651  AD 90 43            LDA LSTDRV
0752  C654  29 01        SA20   AND #1
0753  C656  85 12               STA DRVNUM
0754  C658  4C DC F9            JMP SETLDS
0755  C65B               ;TOGGLE DRVNUM
0756  C65B  A5 12        TOGDRV LDA DRVNUM
0757  C65D  49 01               EOR #1
0758  C65F  29 01               AND #1
0759  C661  85 12               STA DRVNUM
0760  C663  60                  RTS

0762  C664               ;SET PTRS TO ONE FILE STREAM & CHK TYPE
0763  C664  A0 00        FS1SET LDY #0
0764  C666  AD 79 43            LDA F1CNT
0765  C669  CD 7A 43            CMP F2CNT
0766  C66C  F0 16               BEQ FS15
0767  C66E  CE 7A 43            DEC F2CNT
0768  C671  AC 7A 43            LDY F2CNT
0769  C674  B9 7C 43            LDA FILTBL,Y
0770  C677  A8                  TAY
0771  C678  B1 45               LDA (CB),Y
0772  C67A  A0 04               LDY #NTYPES-1
0773  C67C  D9 82 FB     FS10   CMP TYPLST,Y
0774  C67F  F0 03               BEQ FS15
0775  C681  88                  DEY
0776  C682  D0 F8               BNE FS10
0777  C684  98           FS15   TYA
0778  C685  8D 98 43            STA TYPFLG
0779  C688  60                  RTS

0781  C689               ;TEST CHAR IN ACCUM FOR "0" OR "1"
0782  C689  C9 30        TSTOV1 CMP #'0
0783  C68B  F0 06               BEQ TOV1
0784  C68D  C9 31               CMP #'1
0785  C68F  F0 02               BEQ TOV1
0786  C691  09 80               ORA #$80
0787  C693  29 81        TOV1   AND #$81
0788  C695  60                  RTS
0789  C696                      .END
```

```
LINE# LOC   CODE         LINE

0790  C696                          .LIB LOOKUP
```

```
LINE# LOC   CODE        LINE

0792  C696              ;OPTSCH  OPTIMAL SEARCH FOR LOOKUP
0793  C696              ;   AND FNDFIL

0795  C696  A9 00       OPTSCH LDA #0         ;DETERMINE OPTIMAL SEARCH
0796  C698  85 04              STA TEMP       ;INIT DRIVE MASK
0797  C69A  8D 8F 43           STA DRVFLG
0798  C69D  48                 PHA
0799  C69E  AE 7A 43           LDX F2CNT
0800  C6A1  68          OS10   PLA
0801  C6A2  05 04              ORA TEMP
0802  C6A4  48                 PHA
0803  C6A5  A9 01              LDA #1
0804  C6A7  85 04              STA TEMP
0805  C6A9  CA                 DEX
0806  C6AA  30 0F              BMI OS30
0807  C6AC  B5 90              LDA FILDRV,X
0808  C6AE  10 04              BPL OS15
0809  C6B0  06 04              ASL TEMP
0810  C6B2  06 04              ASL TEMP

0812  C6B4  4A          OS15   LSR A
0813  C6B5  90 EA              BCC OS10
0814  C6B7  06 04              ASL TEMP
0815  C6B9  D0 E6              BNE OS10       ;(BRANCH)

0817  C6BB  68          OS30   PLA
0818  C6BC  AA                 TAX
0819  C6BD  BD 0B C7           LDA SCHTBL-1,X
0820  C6C0  48                 PHA
0821  C6C1  29 03              AND #3
0822  C6C3  8D 8E 43           STA DRVCNT
0823  C6C6  68                 PLA
0824  C6C7  0A                 ASL A
0825  C6C8  10 3E              BPL OS40
0826  C6CA  A5 90              LDA FILDRV
0827  C6CC  29 01       OS35   AND #1
0828  C6CE  85 12              STA DRVNUM
0829  C6D0  AD 8E 43           LDA DRVCNT
0830  C6D3  F0 2B              BEQ OS60       ; ONLY 1 DRIVE ADDRESSED
0831  C6D5  20 0C C9           JSR AUTOI      ; CHECK FOR AUTO INIT
0832  C6D8  F0 12              BEQ OS50       ; DRIVE IS ACTIVE
0833  C6DA  20 5B C6           JSR TOGDRV
0834  C6DD  A9 00              LDA #00        ; SET 1 DRIVE ADDRESSED
0835  C6DF  8D 8E 43           STA DRVCNT
0836  C6E2  20 0C C9           JSR AUTOI      ; CHECK FOR AUTO INIT
0837  C6E5  F0 1E              BEQ OS70       ; DRIVE IS ACTIVE
0838  C6E7              OS45
0839  C6E7  A9 74              LDA #NODRIV
0840  C6E9  20 96 C4           JSR CMDERR     ; NO DRIVE ERROR
0841  C6EC              OS50
0842  C6EC  20 5B C6           JSR TOGDRV
0843  C6EF  20 0C C9           JSR AUTOI
0844  C6F2  08                 PHP
0845  C6F3  20 5B C6           JSR TOGDRV
0846  C6F6  28                 PLP
```

```
LINE# LOC   CODE        LINE

0847  C6F7  F0 0C              BEQ OS70
0848  C6F9  A9 00              LDA #00
0849  C6FB  8D 8E 43           STA DRVCNT     ; SET 1 DRIVE ADDRESSED
0850  C6FE  F0 05              BEQ OS70       ; BRA
0851  C700              OS60
0852  C700  20 0C C9           JSR AUTOI
0853  C703  D0 E2              BNE OS45       ; DRIVE NOT ACTIVE
0854  C705              OS70
0855  C705  4C DC F9           JMP SETLDS

0857  C708  2A          OS40   ROL A
0858  C709  4C CC C6           JMP OS35

0860  C70C  00          SCHTBL .BYT 0,$80,$41
0860  C70D  80
0860  C70E  41
0861  C70F  01                 .BYT 1,1,1,1
0861  C710  01
0861  C711  01
0861  C712  01
0862  C713  81                 .BYT $81,$81,$81,$81
0862  C714  81
0862  C715  81
0862  C716  81
0863  C717  42                 .BYT $42,$42,$42,$42
0863  C718  42
0863  C719  42
0863  C71A  42
```

```
LINE# LOC   CODE        LINE

0865  C71B              ; LOOK UP ALL FILES IN STREAM
0866  C71B              ;   AND FILL TABLES W/ INFO

0868  C71B  20 96 C6    LOOKUP JSR OPTSCH
0869  C71E  A9 00       LK05   LDA #0
0870  C720  8D 94 43           STA DELIND
0871  C723  20 78 C8           JSR SRCHST      ;START SEARCH
0872  C726  D0 19              BNE LK25
0873  C728  CE 8E 43    LK10   DEC DRVCNT
0874  C72B  10 01              BPL LK15
0875  C72D  60                 RTS             ;NO MORE DRIVE SEARCHES
0876  C72E  A9 01       LK15   LDA #1           ;TOGGLE DRIVE #
0877  C730  8D 8F 43           STA DRVFLG
0878  C733  20 5B C6           JSR TOGDRV
0879  C736  20 DC F9           JSR SETLDS      ;  TURN ON LED
0880  C739  4C 1E C7           JMP LK05
0881  C73C  20 E6 C8    LK20   JSR SEARCH       ;FIND VALID FN
0882  C73F  F0 10              BEQ LK30        ;END OF SEARCH
0883  C741  20 A4 C7    LK25   JSR COMPAR       ;COMPARE DIR W/ TABLE
0884  C744  AD 91 43           LDA FOUND       ;FOUND FLAG
0885  C747  F0 01              BEQ LK26        ;ALL FN'S NOT FOUND, YET
0886  C749  60                 RTS

0888  C74A  AD 48 43    LK26   LDA ENTFND
0889  C74D  30 ED              BMI LK20
0890  C74F  10 F0              BPL LK25

0892  C751  AD 91 43    LK30   LDA FOUND
0893  C754  F0 D2              BEQ LK10
0894  C756  60                 RTS


0896  C757              ; FIND NEXT FILE NAME MATCHING
0897  C757              ;  ANY FILE IN STREAM & RETURN
0898  C757              ;  WITH ENTRY FOUND STUFFED INTO
0899  C757              ;  TABLES
0900  C757  20 D3 C8    FFRE   JSR SRRE          ;FIND FILE RE-ENTRY
0901  C75A  F0 1A              BEQ FF10
0902  C75C  D0 28              BNE FF25

0904  C75E  A9 01       FF15   LDA #1
0905  C760  8D 8F 43           STA DRVFLG
0906  C763  20 5B C6           JSR TOGDRV
0907  C766  20 DC F9           JSR SETLDS

0909  C769  A9 00       FFST   LDA #0            ;FIND FILE START ENTRY
0910  C76B  8D 94 43           STA DELIND
0911  C76E  20 78 C8           JSR SRCHST
0912  C771  D0 13              BNE FF25
0913  C773  8D 91 43           STA FOUND
0914  C776  AD 91 43    FF10   LDA FOUND
0915  C779  D0 28              BNE FF40
0916  C77B  CE 8E 43           DEC DRVCNT
0917  C77E  10 DE              BPL FF15
0918  C780  60                 RTS
```

```
LINE# LOC   CODE        LINE

0920  C781  20 E6 C8    FNDFIL JSR SEARCH     ;FIND FILE CONTINUOUS...
0921  C784  F0 F0              BEQ FF10      ;... RE-ENTRY, NO CHANNEL ACTI
0922  C786  20 A4 C7    FF25   JSR COMPAR     ;COMPARE FILE NAMES
0923  C789  AE 48 43           LDX ENTFND
0924  C78C  10 07              BPL FF30
0925  C78E  AD 91 43           LDA FOUND
0926  C791  F0 EE              BEQ FNDFIL
0927  C793  D0 0E              BNE FF40

0929  C795  AD 98 43    FF30   LDA TYPFLG
0930  C798  F0 09              BEQ FF40      ;NO TYPE RESTRICTION
0931  C79A  B5 95              LDA PATTYP,X
0932  C79C  29 07              AND #TYPMSK
0933  C79E  CD 98 43           CMP TYPFLG
0934  C7A1  D0 DE              BNE FNDFIL
0935  C7A3  60          FF40   RTS
```

```
LINE# LOC   CODE        LINE

0937  C7A4              ;COMPARE ALL FILENAMES IN STREAM TABLE
0938  C7A4              ;   WITH EACH VALID ENTRY IN THE
0939  C7A4              ;   DIRECTORY.  MATCHES ARE TABULATED

0941  C7A4  A2 FF       COMPAR LDX #$FF
0942  C7A6  8E 48 43           STX ENTFND
0943  C7A9  E8                 INX
0944  C7AA  8E 8C 43           STX PATFLG
0945  C7AD  20 55 C8           JSR CMPCHK
0946  C7B0  F0 06              BEQ CP10
0947  C7B2  60          CP02   RTS              ;ALL ARE FOUND

0949  C7B3  20 60 C8    CP05   JSR CC10
0950  C7B6  D0 FA              BNE CP02
0951  C7B8  A5 12       CP10   LDA DRVNUM
0952  C7BA  55 90              EOR FILDRV,X
0953  C7BC  4A                 LSR A
0954  C7BD  90 0B              BCC CP20        ;RIGHT DRIVE
0955  C7BF  29 40              AND #$40
0956  C7C1  F0 F0              BEQ CP05        ;NO DEFAULT
0957  C7C3  A9 02              LDA #2
0958  C7C5  CD 8E 43           CMP DRVCNT
0959  C7C8  F0 E9              BEQ CP05        ;DON'T USE DEFAULT

0961  C7CA  BD 7C 43    CP20   LDA FILTBL,X     ;GOOD DRIVE MATCH
0962  C7CD  AA                 TAX
0963  C7CE  20 75 C9           JSR FNDLMT
0964  C7D1  A0 03              LDY #3
0965  C7D3  4C E9 C7           JMP CP33
0966  C7D6              CP30
0967  C7D6  BD 00 43           LDA CMDBUF,X
0968  C7D9  D1 27              CMP (DIRBUF)Y
0969  C7DB  F0 0A              BEQ CP32        ;CHARS ARE =
0970  C7DD              ;
0971  C7DD  C9 3F              CMP #'?
0972  C7DF  D0 D2              BNE CP05        ;NO SINGLE PATTERN
0973  C7E1  B1 27              LDA (DIRBUF)Y
0974  C7E3  C9 A0              CMP #$A0
0975  C7E5  F0 CC              BEQ CP05        ;END OF FILENAME
0976  C7E7              CP32
0977  C7E7  E8                 INX
0978  C7E8  C8                 INY
0979  C7E9              CP33
0980  C7E9  EC 78 43           CPX LIMIT
0981  C7EC  B0 09              BCS CP34        ;END OF PATTERN
0982  C7EE              ;
0983  C7EE  BD 00 43           LDA CMDBUF,X
0984  C7F1  C9 2A              CMP #'*
0985  C7F3  F0 0C              BEQ CP40        ;STAR MATCHES ALL
0986  C7F5  D0 DF              BNE CP30        ;KEEP CHECKING
0987  C7F7              CP34
0988  C7F7  C0 13              CPY #19
0989  C7F9  B0 06              BCS CP40        ;END OF FILENAME
0990  C7FB              ;
0991  C7FB  B1 27              LDA (DIRBUF)Y
```

```
LINE# LOC   CODE      LINE


0992  C7FD  C9 A0            CMP #$A0
0993  C7FF  D0 B2            BNE CP05

0995  C801  AE 7B 43  CP40   LDX F2PTR      ;FILENAMES MATCH
0996  C804  8E 48 43         STX ENTFND
0997  C807  B5 95            LDA PATTYP,X   ;STORE INFO IN TABLES
0998  C809  29 80            AND #$80
0999  C80B  8D 8C 43         STA PATFLG
1000  C80E  AD 96 43         LDA INDEX
1001  C811  95 8B            STA ENTIND,X
1002  C813  A5 14            LDA SECTOR
1003  C815  95 86            STA ENTSEC,X
1004  C817  A0 00            LDY #0
1005  C819  B1 27            LDA (DIRBUF),Y
1006  C81B  C8               INY
1007  C81C  48               PHA
1008  C81D  29 40            AND #$40
1009  C81F  85 04            STA TEMP
1010  C821  68               PLA
1011  C822  29 DF            AND #$FF-$20
1012  C824  30 02            BMI CP42
1013  C826            ;
1014  C826  09 20            ORA #$20
1015  C828            CP42
1016  C828  29 27            AND #$27
1017  C82A  05 04            ORA TEMP
1018  C82C  85 04            STA TEMP
1019  C82E  A9 80            LDA #$80
1020  C830  35 95            AND PATTYP,X
1021  C832  05 04            ORA TEMP
1022  C834  95 95            STA PATTYP,X
1023  C836  B5 90            LDA FILDRV,X
1024  C838  29 80            AND #$80
1025  C83A  05 12            ORA DRVNUM
1026  C83C  95 90            STA FILDRV,X
1027  C83E            ;
1028  C83E  B1 27            LDA (DIRBUF),Y
1029  C840  9D 82 43         STA FILTRK,X
1030  C843  C8               INY
1031  C844  B1 27            LDA (DIRBUF),Y
1032  C846  9D 87 43         STA FILSEC,X
1033  C849  AD 4D 43         LDA REC
1034  C84C  D0 07            BNE CP50
1035  C84E  A0 15            LDY #21
1036  C850  B1 27            LDA (DIRBUF)Y
1037  C852  8D 4D 43         STA REC
1038  C855            CP50
1039  C855                                  ;JMP CMPCHK
1040  C855                                  ;RTS

1042  C855            ;CHECK TABLE FOR UNFOUND FILES

1044  C855  A9 FF     CMPCHK LDA #$FF
1045  C857  8D 91 43         STA FOUND
1046  C85A  AD 7A 43         LDA F2CNT
```

```
LINE# LOC   CODE         LINE


1047  C85D  8D 7B 43            STA F2PTR

1049  C860  CE 7B 43     CC10   DEC F2PTR
1050  C863  10 01               BPL CC15
1051  C865  60                  RTS                 ; TABLE EXHAUSTED

1053  C866  AE 7B 43     CC15   LDX F2PTR
1054  C869  B5 95               LDA PATTYP,X
1055  C86B  30 05               BMI CC20
1056  C86D  BD 82 43            LDA FILTRK,X
1057  C870  D0 EE               BNE CC10
1058  C872  A9 00        CC20   LDA #0
1059  C874  8D 91 43            STA FOUND
1060  C877  60                  RTS
```

```
LINE# LOC   CODE        LINE


1062  C878              ;SEARCH DIRECTORY
1063  C878              ;  RETURNS WITH VALID ENTRY W/ DELIND=0
1064  C878              ;  OR RETURNS W/ 1ST DELETED ENTRY
1065  C878              ;  W/ DELIND=1
1066  C878              ;
1067  C878              ;  SRCHST WILL INITIATE A SEARCH
1068  C878              ;  SEARCH WILL CONTINUE A SEARCH

1070  C878              SRCHST
1071  C878  20 0C C9           JSR AUTOI
1072  C87B  A0 00              LDY #0         ;INIT DELETED SECTOR
1073  C87D  8C 93 43           STY DELSEC
1074  C880  88                 DEY
1075  C881  8C 48 43           STY ENTFND

1077  C884  AD 46 FB           LDA DIRTRK     ;START SEARCH AT BEGINNING
1078  C887  85 13              STA TRACK
1079  C889  A9 01              LDA #1
1080  C88B  85 14              STA SECTOR
1081  C88D  8D 95 43           STA LSTBUF
1082  C890  20 71 D7           JSR OPNIRD     ;OPEN INTERNAL READ CHNL

1084  C893  AD 95 43    SR10   LDA LSTBUF      ;LAST BUFFER IF 0
1085  C896  D0 01              BNE SR15
1086  C898  60                 RTS            ;(Z=1)

1088  C899  A9 07       SR15   LDA #7
1089  C89B  8D 97 43           STA FILCNT
1090  C89E  A9 00              LDA #0         ;READ TRACK #
1091  C8A0  20 F4 D7           JSR DRDBYT
1092  C8A3  8D 95 43           STA LSTBUF     ;UPDATE END FLAG

1094  C8A6  20 E6 D7    SR20   JSR GETPNT
1095  C8A9  CE 97 43           DEC FILCNT
1096  C8AC  A0 00              LDY #0
1097  C8AE  B1 27              LDA (DIRBUF),Y ;READ FILE TYPE
1098  C8B0  D0 18              BNE SR30

1100  C8B2  AD 93 43           LDA DELSEC     ;DELETED ENTRY FOUND
1101  C8B5  D0 2F              BNE SEARCH     ;DELETED ENTRY ALREADY FOUN
1102  C8B7  20 7B E6           JSR CURBLK     ;GET CURRENT SECTOR
1103  C8BA  A5 14              LDA SECTOR
1104  C8BC  8D 93 43           STA DELSEC

1106  C8BF  A5 27              LDA DIRBUF     ;GET CURRENT INDEX
1107  C8C1  AE 94 43           LDX DELIND     ;BIT1: WANT DELETED ENTRY
1108  C8C4  8D 94 43           STA DELIND
1109  C8C7  F0 1D              BEQ SEARCH     ;NEED VALID ENTRY
1110  C8C9  60                 RTS            ;(Z=0)

1112  C8CA  A2 01       SR30   LDX #1
1113  C8CC  EC 94 43           CPX DELIND     ;?LOOKING FOR DELETED?
1114  C8CF  D0 2D              BNE SR50       ; NO!
1115  C8D1  F0 13              BEQ SEARCH
```

```
LINE# LOC    CODE         LINE


1117  C8D3   AD 46 FB     SRRE    LDA DIRTRK
1118  C8D6   85 13                STA TRACK
1119  C8D8   AD 92 43             LDA DIRSEC
1120  C8DB   85 14                STA SECTOR
1121  C8DD   20 71 D7             JSR OPNIRD
1122  C8E0   AD 96 43             LDA INDEX
1123  C8E3   20 C6 D7             JSR SETPNT

1125  C8E6   A9 FF        SEARCH  LDA #$FF
1126  C8E8   8D 48 43             STA ENTFND
1127  C8EB   AD 97 43             LDA FILCNT       ;ADJUST FILE COUNT
1128  C8EE   30 08                BMI SR40
1129  C8F0   A9 20                LDA #32          ;INCR BY 32
1130  C8F2   20 3D D5             JSR INCPTR
1131  C8F5   4C A6 C8             JMP SR20

1133  C8F8   20 49 D7     SR40    JSR NXTBUF        ;NEW BUFFER
1134  C8FB   4C 93 C8             JMP SR10         ;(BRANCH)

1136  C8FE   A5 27        SR50    LDA DIRBUF        ;FOUND VALID ENTRY
1137  C900   8D 96 43             STA INDEX        ;SAVE INDEX
1138  C903   20 7B E6             JSR CURBLK       ;GET SECTOR
1139  C906   A5 14                LDA SECTOR
1140  C908   8D 92 43             STA DIRSEC

1142  C90B   60                   RTS              ;(Z=0)
1143  C90C                AUTOI                     ;AUTO-INIT SECTION
1144  C90C   AD F3 10             LDA AUTOFG
1145  C90F   D0 26                BNE AUTO2        ;CHECK DISABLE FLAG
1146  C911   A6 12                LDX DRVNUM
1147  C913   5E A6 10             LSR WPSW,X       ;TEST & CLEAR FLAG
1148  C916   90 1F                BCC AUTO2
1149  C918   A9 FF                LDA #$FF
1150  C91A   8D 9A 43             STA JOBRTN       ; SET ERROR RETURN CODE
1151  C91D   20 A1 D3             JSR ITRIAL       ; INIT SEEK-TEST
1152  C920   A0 FF                LDY #$FF
1153  C922   C9 02                CMP #2
1154  C924   F0 06                BEQ AUTO1        ; NO SYNC = NO DISKETTE
1155  C926   C9 03                CMP #3
1156  C928   F0 02                BEQ AUTO1        ; NO HEADER = NO DIRECTORY
1157  C92A   A0 00                LDY #00
1158  C92C                AUTO1
1159  C92C   A6 12                LDX DRVNUM
1160  C92E   98                   TYA
1161  C92F   9D A1 43             STA NODRV,X      ; SET CONDITION OF NO DRIVE
1162  C932   D0 03                BNE AUTO2
1163  C934   20 B2 D3             JSR INITDR
1164  C937                AUTO2
1165  C937   A6 12                LDX DRVNUM
1166  C939   BD A1 43             LDA NODRV,X
1167  C93C   60                   RTS              ; RETURN NO DRIVE STATUS

1169  C93D                        .END
1170  C93D                        .LIB TRNSFR
```

```
LINE# LOC   CODE        LINE


1172  C93D              ;TRANSFER FILENAME FROM CMD TO BUFFER
1173  C93D              ; A: STRING SIZE
1174  C93D              ; X: STARTING INDEX IN CMDBUF
1175  C93D              ; Y: BUFFER #
1176  C93D  48          TRNAME PHA
1177  C93E  20 75 C9           JSR FNDLMT
1178  C941  20 57 C9           JSR TRCMBF
1179  C944  68                 PLA
1180  C945  38                 SEC
1181  C946  ED 3C 43           SBC STRSIZ
1182  C949  AA                 TAX
1183  C94A  F0 0A              BEQ TN20
1184  C94C  90 08              BCC TN20

1186  C94E  A9 A0              LDA #$A0
1187  C950  91 27       TN10   STA (DIRBUF),Y
1188  C952  C8                 INY
1189  C953  CA                 DEX
1190  C954  D0 FA              BNE TN10
1191  C956  60          TN20   RTS


1193  C957              ;TRANSFER CMD BUFFER TO OTHER BUFFER
1194  C957              ; USES CURRENT BUFFER PTR
1195  C957              ; LIMIT: ENDING INDEX+1 IN CMD BUF
1196  C957              ; X: STARTING INDEX IN CMD BUF
1197  C957              ; Y: BUFFER #

1199  C957  98          TRCMBF TYA
1200  C958  0A                 ASL A
1201  C959  A8                 TAY
1202  C95A  B9 29 00           LDA BUFTAB,Y
1203  C95D  85 27              STA DIRBUF
1204  C95F  B9 2A 00           LDA BUFTAB+1,Y
1205  C962  85 28              STA DIRBUF+1
1206  C964  A0 00              LDY #0
1207  C966              ;
1208  C966  BD 00 43    TR10   LDA CMDBUF,X
1209  C969  91 27              STA (DIRBUF),Y
1210  C96B  C8                 INY
1211  C96C  F0 06              BEQ TR20
1212  C96E  E8                 INX
1213  C96F  EC 78 43           CPX LIMIT
1214  C972  90 F2              BCC TR10
1215  C974  60          TR20   RTS
1216  C975              ;
1217  C975              ;
1218  C975              ;FIND THE LIMIT OF THE STRING IN CMDBUF
1219  C975              ; POINTED TO BY X

1221  C975  A9 00       FNDLMT LDA #0
1222  C977  8D 3C 43           STA STRSIZ
1223  C97A  8A                 TXA
1224  C97B  48                 PHA
1225  C97C              FL05
```

```
LINE# LOC    CODE          LINE


1226  C97C   BD 00 43             LDA CMDBUF,X
1227  C97F   C9 2C                CMP #',
1228  C981   F0 14                BEQ FL10
1229  C983   C9 3D                CMP #'=
1230  C985   F0 10                BEQ FL10
1231  C987   EE 3C 43             INC STRSIZ
1232  C98A   E8                   INX
1233  C98B   A9 0F                LDA #15
1234  C98D   CD 3C 43             CMP STRSIZ
1235  C990   90 05                BCC FL10
1236  C992   EC 76 43             CPX CMDSIZ
1237  C995   90 E5                BCC FL05
1238  C997   8E 78 43      FL10   STX LIMIT
1239  C99A   68                   PLA
1240  C99B   AA                   TAX
1241  C99C   60                   RTS
```

```
LINE# LOC   CODE        LINE


1244  C99D              ; GET FILE ENTRY FROM DIRECTORY
1245  C99D              ; CALLED BY STDIR, GETDIR
1246  C99D  A5 16       GETNAM LDA SA          ;SAVE VARIABLES
1247  C99F  48                 PHA
1248  C9A0  A5 15              LDA LINDX
1249  C9A2  48                 PHA
1250  C9A3  20 AD C9           JSR GNSUB
1251  C9A6  68                 PLA            ;RESTORE VARIABLES
1252  C9A7  85 15              STA LINDX
1253  C9A9  68                 PLA
1254  C9AA  85 16              STA SA
1255  C9AC  60                 RTS

1257  C9AD  A9 11       GNSUB  LDA #IRSA
1258  C9AF  85 16              STA SA
1259  C9B1  20 64 D4           JSR FNDRCH
1260  C9B4  20 E6 D7           JSR GETPNT
1261  C9B7  AD 48 43           LDA ENTFND
1262  C9BA  10 0C              BPL GN05       ;MORE FILES
1263  C9BC  AD 8F 43           LDA DRVFLG
1264  C9BF  D0 0C              BNE GN050
1265  C9C1  20 D5 CA           JSR MSGFRE     ;SEND BLOCKS FREE
1266  C9C4  18                 CLC            ;(C=0): END
1267  C9C5  4C 7C CA           JMP GN45       ;TERMINATE

1269  C9C8  AD 8F 43    GN05   LDA DRVFLG      ;(DRVFLG=0):
1270  C9CB  F0 1F              BEQ GN10       ;  SEND FILE NAME
1271  C9CD              GN050
1272  C9CD  CE 8F 43           DEC DRVFLG     ;(DRVFLG=-1):NEW DIR
1273  C9D0  D0 0D              BNE GN051
1274  C9D2  CE 8F 43           DEC DRVFLG     ;(DRVFLG=-1):
1275  C9D5  20 5B C6           JSR TOGDRV     ;  SEND BLOCKS FREE
1276  C9D8  20 D5 CA           JSR MSGFRE
1277  C9DB  38                 SEC
1278  C9DC  4C 5B C6           JMP TOGDRV

1280  C9DF  A9 00       GN051  LDA #0
1281  C9E1  8D 75 43           STA NBTEMP+1
1282  C9E4  8D 8F 43           STA DRVFLG     ;RESET FLAG
1283  C9E7  20 88 CA           JSR NEWDIR
1284  C9EA  38                 SEC
1285  C9EB  60                 RTS

1287  C9EC  A2 18       GN10   LDX #DIRLEN     ;SET NUMBER BLOCKS
1288  C9EE  A0 1D              LDY #29        ; & ADJUST SPACING
1289  C9F0  B1 27              LDA (DIRBUF),Y
1290  C9F2  8D 75 43           STA NBTEMP+1
1291  C9F5  F0 02              BEQ GN12
1292  C9F7  A2 16              LDX #DIRLEN-2
1293  C9F9  88          GN12   DEY
1294  C9FA  B1 27              LDA (DIRBUF),Y
1295  C9FC  8D 74 43           STA NBTEMP
1296  C9FF  E0 16              CPX #DIRLEN-2
1297  CA01  F0 0A              BEQ GN14
1298  CA03  C9 0A              CMP #10
```

```
LINE# LOC   CODE        LINE

1299  CA05  90 06             BCC GN14
1300  CA07  CA                DEX
1301  CA08  C9 64             CMP #100
1302  CA0A  90 01             BCC GN14
1303  CA0C  CA                DEX

1305  CA0D  20 7D CA    GN14  JSR BLKNB        ;CLEAR NAME BUFFER

1307  CA10  B1 27             LDA (DIRBUF),Y ;SET TYPE CHARS
1308  CA12  48                PHA
1309  CA13  0A                ASL A            ;(USED IN BCS)
1310  CA14  10 05             BPL GN15
1311  CA16  A9 3C             LDA #'<
1312  CA18  9D B9 43          STA NAMBUF+1,X
1313  CA1B              GN15
1314  CA1B  68                PLA
1315  CA1C  29 0F             AND #$F
1316  CA1E  A8                TAY
1317  CA1F  B9 8C FB          LDA TP2LST,Y
1318  CA22  9D B8 43          STA NAMBUF,X
1319  CA25  CA                DEX
1320  CA26  B9 87 FB          LDA TP1LST,Y
1321  CA29  9D B8 43          STA NAMBUF,X
1322  CA2C  CA                DEX
1323  CA2D  B9 82 FB          LDA TYPLST,Y
1324  CA30  9D B8 43          STA NAMBUF,X
1325  CA33  CA                DEX
1326  CA34  CA                DEX
1327  CA35  B0 05             BCS GN20         ;(FROM ASL)
1328  CA37  A9 2A             LDA #'*          ;FILE NOT CLOSED
1329  CA39  9D B9 43          STA NAMBUF+1,X
1330  CA3C  A9 A0       GN20  LDA #$A0
1331  CA3E  9D B8 43          STA NAMBUF,X
1332  CA41  CA                DEX
1333  CA42  A0 12             LDY #18
1334  CA44  B1 27       GN22  LDA (DIRBUF),Y
1335  CA46  9D B8 43          STA NAMBUF,X
1336  CA49  CA                DEX
1337  CA4A  88                DEY
1338  CA4B  C0 03             CPY #3
1339  CA4D  B0 F5             BCS GN22

1341  CA4F  A9 22             LDA #'"          ;SEND NAME IN QUOTES
1342  CA51  9D B8 43          STA NAMBUF,X
1343  CA54  E8          GN30  INX
1344  CA55  E0 20             CPX #$20
1345  CA57  B0 0B             BCS GN35
1346  CA59  BD B8 43          LDA NAMBUF,X
1347  CA5C  C9 22             CMP #'"
1348  CA5E  F0 04             BEQ GN35
1349  CA60  C9 A0             CMP #$A0
1350  CA62  D0 F0             BNE GN30
1351  CA64  A9 22       GN35  LDA #'"
1352  CA66  9D B8 43          STA NAMBUF,X
1353  CA69  E8          GN37  INX
```

```
LOC   CODE        LINE

CA6A  E0 20               CPX #$20
CA6C  B0 0A               BCS GN40
CA6E  A9 7F               LDA #$7F
CA70  3D B8 43            AND NAMBUF,X
CA73  9D B8 43            STA NAMBUF,X
CA76  10 F1               BPL GN37

CA78  20 81 C7    GN40    JSR FNDFIL
CA7B  38                  SEC
CA7C  60          GN45    RTS

CA7D  A0 1B       BLKNB   LDY #NBSIZ        ;BLANK NAMBUF
CA7F  A9 20               LDA #$20
CA81  99 B7 43    BLKNB1  STA NAMBUF-1,Y
CA84  88                  DEY
CA85  D0 FA               BNE BLKNB1
CA87  60                  RTS

CA88              ;NEW DIRECTORY IN LISTING
CA88              NEWDIR
CA88  20 0C C9            JSR AUTOI
CA8B  20 C5 F9            JSR RDDIR
CA8E  20 7D CA            JSR BLKNB
CA91  A9 FF               LDA #$FF
CA93  85 04               STA TEMP
CA95  A6 12               LDX DRVNUM
CA97  8E 74 43            STX NBTEMP
CA9A  A9 00               LDA #0
CA9C  8D 75 43            STA NBTEMP+1
CA9F  BD 96 FB            LDA IPBM,X
CAA2  85 28               STA DIRBUF+1
CAA4  AD 49 FB            LDA DSKNAM
CAA7  85 27               STA DIRBUF
CAA9  A0 16               LDY #22

CAAB  B1 27       ND10    LDA (DIRBUF),Y
CAAD  C9 A0               CMP #$A0
CAAF  D0 0B               BNE ND20
CAB1  A9 31               LDA #'1          ;VERSION # 1
CAB3  2C                  .BYTE $2C
CAB4              ND15
CAB4  B1 27               LDA (DIRBUF)Y
CAB6  C9 A0               CMP #$A0
CAB8  D0 02               BNE ND20
CABA              ;
CABA  A9 20               LDA #$20
CABC              ND20
CABC  99 BA 43            STA NAMBUF+2,Y
CABF  88                  DEY
CAC0  10 F2               BPL ND15
CAC2  A9 12               LDA #$12
CAC4  8D B8 43            STA NAMBUF
CAC7  A9 22               LDA #'"
CAC9  8D B9 43            STA NAMBUF+1
CACC  8D CA 43            STA NAMBUF+18
```

```
LINE# LOC   CODE        LINE


1409  CACF  A9 20                LDA #$20
1410  CAD1  8D CB 43             STA NAMBUF+19
1411  CAD4  60                   RTS

1413  CAD5  20 7D CA    MSGFRE JSR BLKNB
1414  CAD8  A0 0B                LDY #MSGLEN-1
1415  CADA  B9 E6 CA    MSG1     LDA FREMSG,Y
1416  CADD  99 B8 43             STA NAMBUF,Y
1417  CAE0  88                   DEY
1418  CAE1  10 F7                BPL MSG1
1419  CAE3  4C 68 F8             JMP NUMFRE       ; GET NUMBER OF BYTES

1421  CAE6  42 4C       FREMSG .BYTE 'BLOCKS FREE. '
1422  CAF2              MSGLEN =*-FREMSG
1423  CAF2                     .END
1424  CAF2                     .LIB SCRTCH
```

```
LINE# LOC   CODE          LINE

 1426  CAF2               ; SCRATCH FILE(S)
 1427  CAF2  20 64 C6     SCRTCH JSR FS1SET         ;SET UP FOR 1 STREAM
 1428  CAF5  20 ED C5            JSR ALLDRS
 1429  CAF8  20 96 C6            JSR OPTSCH
 1430  CAFB  A9 00               LDA #0
 1431  CAFD  85 19               STA R0             ;USED AS FILE COUNT
 1432  CAFF  20 69 C7            JSR FFST
 1433  CB02  30 3D               BMI SC30
 1434  CB04  20 F8 E5     SC15   JSR TSTCHN          ;IS IT ACTIVE ?
 1435  CB07  90 33               BCC SC25           ;YES - DONT SCRATCH
 1436  CB09  A0 00               LDY #0
 1437  CB0B  B1 27               LDA (DIRBUF),Y
 1438  CB0D  29 40               AND #$40           ;LOCK BIT
 1439  CB0F  D0 2B               BNE SC25           ;IT'S LOCKED
 1440  CB11               ;
 1441  CB11  20 74 CB            JSR DELDIR         ;DELETE DIRECTORY
 1442  CB14  A0 13               LDY #19            ;IS THIS A RELATIVE ?
 1443  CB16  B1 27               LDA (DIRBUF),Y     ;HAS A SS ?
 1444  CB18  F0 0A               BEQ SC17           ;NO
 1445  CB1A  85 13               STA TRACK          ;YES - SAVE TRACK
 1446  CB1C  C8                  INY
 1447  CB1D  B1 27               LDA (DIRBUF),Y     ;GET SECTOR
 1448  CB1F  85 14               STA SECTOR
 1449  CB21  20 4C CB            JSR DELFIL         ;DELETE BY LINKS
 1450  CB24  AE 48 43     SC17   LDX ENTFND
 1451  CB27  A9 20               LDA #$20
 1452  CB29  35 95               AND PATTYP,X
 1453  CB2B  D0 0D               BNE SC20           ;CREATED, NOT CLOSED

 1455  CB2D  BD 82 43            LDA FILTRK,X       ;DELETE BY LINKS
 1456  CB30  85 13               STA TRACK
 1457  CB32  BD 87 43            LDA FILSEC,X
 1458  CB35  85 14               STA SECTOR
 1459  CB37  20 4C CB            JSR DELFIL

 1461  CB3A  E6 19        SC20   INC R0
 1462  CB3C  20 57 C7     SC25   JSR FFRE
 1463  CB3F  10 C3               BPL SC15

 1465  CB41  A5 19        SC30   LDA R0              ;FINISHED, SET
 1466  CB43  85 13               STA TRACK          ;  FILE COUNT
 1467  CB45  A9 01               LDA #1
 1468  CB47  A0 00               LDY #0
 1469  CB49  4C 70 C4            JMP SCREND         ;END OF SCRATCH

 1471  CB4C  20 77 F8     DELFIL JSR FRETS           ;DELETE FILE BY LINKS
 1472  CB4F  20 71 D7            JSR OPNIRD         ;UPDATE BAM
 1473  CB52  A9 00        DEL2   LDA #0
 1474  CB54  20 C6 D7            JSR SETPNT
 1475  CB57  20 CD D4            JSR RDBYT
 1476  CB5A  85 13               STA TRACK
 1477  CB5C  20 CD D4            JSR RDBYT
 1478  CB5F  85 14               STA SECTOR
 1479  CB61  A5 13               LDA TRACK
 1480  CB63  D0 06               BNE DEL1
```

```
LINE# LOC   CODE        LINE


1481  CB65  20 32 F8              JSR MAPOUT
1482  CB68  4C 9D D5              JMP FRECHN
1483  CB6B  20 77 F8    DEL1      JSR FRETS
1484  CB6E  20 49 D7              JSR NXTBUF
1485  CB71  4C 52 CB              JMP DEL2

1487  CB74  A0 00       DELDIR LDY #0            ;DELETE DIR ENTRY
1488  CB76  98                    TYA
1489  CB77  91 27                 STA (DIRBUF),Y
1490  CB79  20 A0 E6              JSR WRTOUT
1491  CB7C  4C AF D8              JMP WATJOB

1493  CB7F                        .END
1494  CB7F                        .LIB DUPLCT
```

```
LINE# LOC   CODE        LINE


1496  CB7F              ; DUPLICATE DISK

1498  CB7F              DUPLCT
1499  CB7F  20 DE CC           JSR PRSEQ
1500  CB82  A5 91              LDA FILDRV+1
1501  CB84  85 12              STA DRVNUM
1502  CB86  A9 18              LDA #LED0+LED1
1503  CB88  0D 82 02           ORA PBD2
1504  CB8B  8D 82 02           STA PBD2
1505  CB8E  20 B2 D3           JSR INITDR      ;INIT SRC
1506  CB91              ;
1507  CB91  20 C8 FA           JSR SETBMP
1508  CB94  A0 02              LDY #2
1509  CB96  B1 02              LDA (BMPNT)Y
1510  CB98  CD 9F 10           CMP VERNUM
1511  CB9B  F0 03              BEQ DUP1
1512  CB9D  4C 87 D8           JMP VNERR
1513  CBA0              DUP1
1514  CBA0  20 5B C6           JSR TOGDRV
1515  CBA3  0A                 ASL A
1516  CBA4  A8                 TAY
1517  CBA5  49 02              EOR #2
1518  CBA7  AA                 TAX
1519  CBA8  BD 42 43           LDA DSKID,X
1520  CBAB  99 42 43           STA DSKID,Y
1521  CBAE  BD 43 43           LDA DSKID+1,X
1522  CBB1  99 43 43           STA DSKID+1,Y
1523  CBB4              ;
1524  CBB4  20 C8 FA           JSR SETBMP
1525  CBB7  A0 02              LDY #2
1526  CBB9  AD 9F 10           LDA VERNUM
1527  CBBC  91 02              STA (BMPNT)Y
1528  CBBE              ;
1529  CBBE  20 59 D6           JSR CLDCHN
1530  CBC1  A9 01              LDA #1
1531  CBC3  85 13              STA TRACK
1532  CBC5  20 50 CC           JSR FORMAT      ;FORMAT DST DRIVE

1534  CBC8              ; COPY BLOCKS FROM ONE DRIVE TO OTHER
1535  CBC8  A5 13       CPYD1  LDA TRACK
1536  CBCA  20 2C FB           JSR MAXSEC
1537  CBCD  85 14              STA SECTOR
1538  CBCF  C6 14              DEC SECTOR
1539  CBD1  20 E3 CB           JSR CPYTRK      ;COPY ONE TRACK
1540  CBD4  E6 13              INC TRACK
1541  CBD6  A5 13              LDA TRACK
1542  CBD8  CD B0 10           CMP MAXTRK
1543  CBDB  D0 EB              BNE CPYD1
1544  CBDD  20 B2 D3           JSR INITDR
1545  CBE0  4C 66 C4           JMP ENDCMD

1547  CBE3              ; COPY ONE TRACK
1548  CBE3  20 F1 CB    CPYTRK JSR SETRH
1549  CBE6  20 0C CC           JSR READS       ;READ 10 SECTORS
1550  CBE9  20 30 CC           JSR WRITES      ;WRITE 10 SECTORS
```

```
LINE# LOC   CODE        LINE


1551  CBEC  A5 14              LDA SECTOR
1552  CBEE  10 F3              BPL CPYTRK
1553  CBF0  60                 RTS

1555  CBF1              ;
1556  CBF1              ;
1557  CBF1  A5 12       SETRH  LDA DRVNUM
1558  CBF3  49 01              EOR #1
1559  CBF5  8D 3E 43           STA CMD
1560  CBF8  A9 0A       SETR5  LDA #10
1561  CBFA  85 06              STA TEMP+2
1562  CBFC  A5 06       SETR3  LDA TEMP+2
1563  CBFE  20 DC D9           JSR SETH
1564  CC01  C6 14              DEC SECTOR
1565  CC03  30 06              BMI SETR6
1566  CC05  C6 06              DEC TEMP+2
1567  CC07  10 F3              BPL SETR3
1568  CC09  E6 06              INC TEMP+2
1569  CC0B  60          SETR6  RTS

1571  CC0C              ; READ TEMP+2 BLOCKS IN
1572  CC0C  AD 3E 43    READS  LDA CMD
1573  CC0F  09 80              ORA #READ
1574  CC11  8D 3E 43           STA CMD
1575  CC14  A6 06              LDX TEMP+2
1576  CC16              READS1
1577  CC16  AD 3E 43           LDA CMD
1578  CC19  20 1B D8           JSR SETJOB
1579  CC1C              ;
1580  CC1C  E0 0A              CPX #10
1581  CC1E  F0 03              BEQ READS8
1582  CC20  E8                 INX
1583  CC21  D0 F3              BNE READS1
1584  CC23  A6 06       READS8 LDX TEMP+2
1585  CC25              READS3
1586  CC25  20 AF D8           JSR WATJOB
1587  CC28  E0 0A       READ10 CPX #10
1588  CC2A  F0 03              BEQ READ15
1589  CC2C  E8                 INX
1590  CC2D  D0 F6              BNE READS3
1591  CC2F  60          READ15 RTS

1593  CC30              ; WRITE TEMP+2 BUFFERS OUT
1594  CC30  A9 90       WRITES LDA #WRITE
1595  CC32  05 12              ORA DRVNUM
1596  CC34  8D 3E 43           STA CMD
1597  CC37  A6 06              LDX TEMP+2
1598  CC39              WRITO
1599  CC39  20 1B D8           JSR SETJOB
1600  CC3C  E0 0A       WRIT30 CPX #10
1601  CC3E  F0 03              BEQ WRIT5
1602  CC40  E8                 INX
1603  CC41  D0 F6              BNE WRITO
1604  CC43              WRIT5
1605  CC43  A6 06              LDX TEMP+2
```

```
LINE# LOC   CODE        LINE


1606  CC45              WRIT10
1607  CC45  20 AF D8           JSR WATJOB
1608  CC48  E0 0A       WRIT15 CPX #10
1609  CC4A  F0 03              BEQ WRIT20
1610  CC4C  E8                 INX
1611  CC4D  D0 F6              BNE WRIT10
1612  CC4F  60          WRIT20 RTS

1614  CC50              ; TRANSFER FORMAT CODE TO BUFFER 0
1615  CC50              ;   & START CONTROLLER FORMATTING
1616  CC50              FORMAT
1617  CC50  A0 00              LDY #0
1618  CC52  B9 00 C0    FMT102 LDA CODE,Y
1619  CC55  99 00 1D           STA FBUFS,Y
1620  CC58  B9 00 C1           LDA CODE+$100,Y
1621  CC5B  99 00 1E           STA FBUFS+$100,Y
1622  CC5E  B9 00 C2           LDA CODE+$200,Y
1623  CC61  99 00 1F           STA FBUFS+$200,Y
1624  CC64  B9 00 C3           LDA CODE+$300,Y
1625  CC67  99 00 20           STA FBUFS+$300,Y
1626  CC6A  C8                 INY
1627  CC6B  D0 E5              BNE FMT102
1628  CC6D              FMT104
1629  CC6D  A9 00              LDA #0
1630  CC6F  20 DC D9           JSR SETH
1631  CC72  A5 12              LDA DRVNUM
1632  CC74  09 E0              ORA #EXEC
1633  CC76  8D 03 10           STA JOBS
1634  CC79  AD 03 10    FMT105 LDA JOBS
1635  CC7C  30 FB              BMI FMT105
1636  CC7E  C9 01              CMP #1
1637  CC80  F0 07              BEQ FMT110
1638  CC82  A9 03              LDA #3
1639  CC84  A2 00              LDX #0
1640  CC86  4C B3 EE           JMP ERROR
1641  CC89  60          FMT110 RTS
1642  CC8A              ;
1643  CC8A              ;
1644  CC8A                     .END
1645  CC8A                     .LIB COPYSETUP
```

```
LINE# LOC   CODE        LINE


1647  CC8A              ;
1648  CC8A              ; DSKCPY CHECK FOR TYPE
1649  CC8A              ; AND PARSES SPECIAL CASE
1650  CC8A              ;
1651  CC8A              DSKCPY
1652  CC8A  20 B3 C4           JSR PRSCLN     ;FIND ":"
1653  CC8D  D0 1D              BNE DX0000
1654  CC8F              ;
1655  CC8F  20 DE CC           JSR PRSEQ
1656  CC92              ;
1657  CC92  A9 2A              LDA #'*        ;CPY ALL
1658  CC94  A2 27              LDX #39        ;PUT AT BUFFER END
1659  CC96  8E 7D 43           STX FILTBL+1
1660  CC99  9D 00 43           STA CMDBUF,X   ;PLACE *
1661  CC9C  E8                 INX
1662  CC9D  8E 76 43           STX CMDSIZ
1663  CCA0  A2 01              LDX #1         ;SET UP CNT'S
1664  CCA2  8E 79 43           STX F1CNT
1665  CCA5  E8                 INX
1666  CCA6  8E 7A 43           STX F2CNT
1667  CCA9  4C 21 CD           JMP MOVLP2     ;ENTER ROUTINE
1668  CCAC              ;
1669  CCAC  20 C6 C4    DX0000 JSR TC30        ;NORMAL PARSE
1670  CCAF  20 ED C5    DX0005 JSR ALLDRS      ;PUT DRV'S IN FILTBL
1671  CCB2  AD 8D 43           LDA IMAGE      ;GET PARSE IMAGE
1672  CCB5  29 55              AND #%01010101 ;VAL FOR PATT COPY
1673  CCB7  D0 1B              BNE DX0020     ;MUST BE CONCAT OR NORMAL
1674  CCB9  AE 7C 43           LDX FILTBL     ;CHK FOR *
1675  CCBC  BD 00 43           LDA CMDBUF,X
1676  CCBF  C9 2A              CMP #'*
1677  CCC1  D0 11              BNE DX0020
1678  CCC3  A2 01              LDX #1         ;SET CNT'S
1679  CCC5  8E 79 43           STX F1CNT
1680  CCC8  E8                 INX
1681  CCC9  8E 7A 43           STX F2CNT
1682  CCCC  4C 05 CD           JMP CPYDTD     ;GO COPY
1683  CCCF  A9 30       DX0010 LDA #BADSYN     ;SYNTAX ERROR
1684  CCD1  4C 96 C4           JMP CMDERR
1685  CCD4  AD 8D 43    DX0020 LDA IMAGE       ;CHK FOR NORMAL
1686  CCD7  29 D9              AND #%11011001
1687  CCD9  D0 F4              BNE DX0010
1688  CCDB  4C C4 CD           JMP COPY
1689  CCDE              ;
1690  CCDE              PRSEQ
1691  CCDE  A9 3D              LDA #'=        ;SPECIAL CASE
1692  CCE0  20 36 C5           JSR PARSE
1693  CCE3  D0 05              BNE X0020
1694  CCE5  A9 30       X0015  LDA #BADSYN
1695  CCE7  4C 96 C4           JMP CMDERR
1696  CCEA  B9 00 43    X0020  LDA CMDBUF,Y
1697  CCED  20 89 C6           JSR TSTOV1
1698  CCF0  30 F3              BMI X0015
1699  CCF2  85 91              STA FILDRV+1   ;SRC DRV
1700  CCF4  88                 DEY
1701  CCF5  88                 DEY
```

```
LINE# LOC   CODE        LINE

1702  CCF6  B9 00 43          LDA CMDBUF,Y
1703  CCF9  20 89 C6          JSR TSTOV1
1704  CCFC  30 E7             BMI X0015
1705  CCFE  C5 91             CMP FILDRV+1   ;CANNOT BE EQUAL
1706  CD00  F0 E3             BEQ X0015
1707  CD02  85 90             STA FILDRV     ;DEST DRV
1708  CD04  60                RTS
1709  CD05                    .END
1710  CD05                    .LIB COPYALL
```

```
LINE# LOC   CODE         LINE

1712  CD05               ;
1713  CD05               ; COPY DISK TO DISK ROUTINES
1714  CD05               ;
1715  CD05  AD 7D 43     CPYDTD LDA FILTBL+1     ;SAVE IN TEMP
1716  CD08  85 04               STA TEMP
1717  CD0A  A0 28               LDY #40          ;40 CHAR BUFFER
1718  CD0C  AE 76 43            LDX CMDSIZ       ;PREP TO MOVE
1719  CD0F  8C 76 43            STY CMDSIZ       ;END OF FILENAME2
1720  CD12  88           MOVLP1 DEY
1721  CD13  CA                  DEX
1722  CD14  BD 00 43            LDA CMDBUF,X     ;MOV FN LIFO
1723  CD17  99 00 43            STA CMDBUF,Y
1724  CD1A  E4 04               CPX TEMP         ;ACTUAL F2 VAL
1725  CD1C  D0 F4               BNE MOVLP1
1726  CD1E  8C 7D 43            STY FILTBL+1     ;POINTER TO F2
1727  CD21  20 96 C6     MOVLP2 JSR OPTSCH
1728  CD24  20 A7 CD            JSR PUPS1        ;SETUP FIRST PASS
1729  CD27  20 69 C7            JSR FFST         ;FIRST MATCH
1730  CD2A  10 2D               BPL FIXIT        ;ENTRY FOUND?
1731  CD2C  30 28               BMI ENDIT        ;NO
1732  CD2E               ;
1733  CD2E  68           EXLP0  PLA               ;PULL NEEDED VARS
1734  CD2F  8D 92 43            STA DIRSEC
1735  CD32  68                  PLA
1736  CD33  8D 7D 43            STA FILTBL+1
1737  CD36  68                  PLA
1738  CD37  8D 95 43            STA LSTBUF
1739  CD3A  68                  PLA
1740  CD3B  8D 97 43            STA FILCNT
1741  CD3E  68                  PLA
1742  CD3F  8D 96 43            STA INDEX
1743  CD42  68                  PLA
1744  CD43  8D 91 43            STA FOUND
1745  CD46  68                  PLA
1746  CD47  8D 94 43            STA DELIND
1747  CD4A  68                  PLA
1748  CD4B  8D 8F 43            STA DRVFLG
1749  CD4E               ;
1750  CD4E  20 A7 CD     EXLP1  JSR PUPS1         ;SET UP VARS
1751  CD51  20 57 C7            JSR FFRE         ;NEXT MATCH
1752  CD54  10 03               BPL FIXIT        ;FOUND ONE?
1753  CD56  4C 66 C4     ENDIT  JMP ENDCMD        ;NO! SO BYE
1754  CD59               ;
1755  CD59  AD 8F 43     FIXIT  LDA DRVFLG        ;PUSH NEEDED VARS
1756  CD5C  48                  PHA
1757  CD5D  AD 94 43            LDA DELIND
1758  CD60  48                  PHA
1759  CD61  AD 91 43            LDA FOUND
1760  CD64  48                  PHA
1761  CD65  AD 96 43            LDA INDEX
1762  CD68  48                  PHA
1763  CD69  AD 97 43            LDA FILCNT
1764  CD6C  48                  PHA
1765  CD6D  AD 95 43            LDA LSTBUF
1766  CD70  48                  PHA
```

```
LINE# LOC   CODE        LINE

1767  CD71  AD 7D 43           LDA FILTBL+1
1768  CD74  48                 PHA
1769  CD75  AD 92 43           LDA DIRSEC
1770  CD78  48                 PHA
1771  CD79              ;
1772  CD79  20 97 CD    EXLP2  JSR TRFNME       ;TRANSFER NAME
1773  CD7C  A9 01              LDA #1           ;FAKE OUT LOOKUP
1774  CD7E  8D 79 43           STA F1CNT
1775  CD81  8D 7A 43           STA F2CNT
1776  CD84  20 1B C7           JSR LOOKUP
1777  CD87  A9 01              LDA #1
1778  CD89  8D 79 43           STA F1CNT
1779  CD8C  A9 02              LDA #2           ;REAL
1780  CD8E  8D 7A 43           STA F2CNT
1781  CD91  20 16 CE           JSR CY           ;COPY IT
1782  CD94  4C 2E CD           JMP EXLP0        ;NEXT ONE FOLKS
1783  CD97              ;
1784  CD97              ;  TRANSFER NAME (DIRBUF) TO CMDBUF
1785  CD97              ;
1786  CD97  A0 03       TRFNME LDY #3           ;BOTH INDEXES
1787  CD99  8C 7C 43           STY FILTBL       ;BEGINING OF FILENAME1
1788  CD9C  B1 27       TRF0   LDA (DIRBUF)Y    ;MOVE IT
1789  CD9E  99 00 43           STA CMDBUF,Y
1790  CDA1  C8                 INY
1791  CDA2  C0 13              CPY #19          ;ALL 16 CHARS PASSED?
1792  CDA4  D0 F6              BNE TRF0
1793  CDA6  60                 RTS
1794  CDA7              ;
1795  CDA7              ;  SET UP SUBROUTINE
1796  CDA7              ;
1797  CDA7  A9 00       PUPS1  LDA #0
1798  CDA9  8D 8E 43           STA DRVCNT
1799  CDAC  8D 82 43           STA FILTRK
1800  CDAF  8D 83 43           STA FILTRK+1
1801  CDB2  A5 91              LDA FILDRV+1     ;GET DRIVE NUMBER
1802  CDB4  29 01              AND #1           ;ONLY
1803  CDB6  85 12              STA DRVNUM
1804  CDB8  09 01              ORA #1
1805  CDBA  8D 93 43           STA DELSEC       ;NONZERO
1806  CDBD  AD 7D 43           LDA FILTBL+1     ;FN1=FN2
1807  CDC0  8D 7C 43           STA FILTBL
1808  CDC3  60                 RTS
1809  CDC4              ;
1810  CDC4                     .END
1811  CDC4                     .LIB RENAME
```

```
LINE# LOC   CODE        LINE


1813  CDC4              ; COPY FILE(S) TO ONE FILE
1814  CDC4              ;
1815  CDC4              COPY                           ;FILENAMES, OPTIMIZE
1816  CDC4  20 1B C7           JSR LOOKUP     ;LOOK IP ALL FILES
1817  CDC7  AD 7A 43           LDA F2CNT
1818  CDCA  C9 03              CMP #3
1819  CDCC  90 42              BCC COP10
1820  CDCE              ;
1821  CDCE  A5 90              LDA FILDRV
1822  CDD0  C5 91              CMP FILDRV+1
1823  CDD2  D0 3C              BNE COP10
1824  CDD4              ;
1825  CDD4  A5 8B              LDA ENTIND
1826  CDD6  C5 8C              CMP ENTIND+1
1827  CDD8  D0 36              BNE COP10
1828  CDDA              ;
1829  CDDA  A5 86              LDA ENTSEC
1830  CDDC  C5 87              CMP ENTSEC+1
1831  CDDE  D0 30              BNE COP10
1832  CDE0              ;
1833  CDE0  20 F6 CE           JSR CHKIN      ;CONCAT
1834  CDE3  A9 01              LDA #1
1835  CDE5  8D 7B 43           STA F2PTR
1836  CDE8  20 5A CE           JSR OPIRFL
1837  CDEB              ;
1838  CDEB  20 9C D4           JSR TYPFIL
1839  CDEE  F0 04              BEQ COP01
1840  CDF0  C9 02              CMP #PRGTYP
1841  CDF2  D0 05              BNE COP05
1842  CDF4              COP01
1843  CDF4  A9 64              LDA #MISTYP
1844  CDF6  20 96 C4           JSR CMDERR
1845  CDF9              COP05
1846  CDF9  A9 12              LDA #IWSA
1847  CDFB  85 16              STA SA
1848  CDFD  A5 BD              LDA LINTAB+IRSA
1849  CDFF  85 BE              STA LINTAB+IWSA
1850  CE01  A9 FF              LDA #$FF
1851  CE03  85 BD              STA LINTAB+IRSA
1852  CE05  20 66 E2           JSR APPEND
1853  CE08  A2 02              LDX #2
1854  CE0A  20 28 CE           JSR CY10
1855  CE0D  4C 66 C4           JMP ENDCMD
1856  CE10              COP10
1857  CE10  20 16 CE           JSR CY
1858  CE13  4C 66 C4           JMP ENDCMD
1859  CE16              ;
1860  CE16              ;
1861  CE16              CY
1862  CE16  20 10 CF           JSR CHKIO      ;CHECK FILES FOR EXISTENCE

1864  CE19  A5 90              LDA FILDRV
1865  CE1B  29 01              AND #1
1866  CE1D  85 12              STA DRVNUM
1867  CE1F  20 81 D7           JSR OPNIWR     ;OPEN INTERNAL WRITE CHNL
```

```
LINE# LOC   CODE        LINE


1868  CE22  20 FB D9           JSR ADDFIL      ;ADD TO DIRECTORY
1869  CE25  AE 79 43           LDX F1CNT

1871  CE28  8E 7B 43    CY10   STX F2PTR        ;SET UP READ FILE
1872  CE2B  20 5A CE           JSR OPIRFL
1873  CE2E  A9 08              LDA #EOISND
1874  CE30  85 AA              STA EOIFLG
1875  CE32  4C 38 CE           JMP CY20
1876  CE35              CY15
1877  CE35  20 27 D3           JSR PIBYTE
1878  CE38              CY20
1879  CE38  20 94 CE           JSR GIBYTE
1880  CE3B  A9 80              LDA #LRF
1881  CE3D  20 E7 E5           JSR TSTFLG
1882  CE40  F0 F3              BEQ CY15
1883  CE42              ;
1884  CE42  20 9C D4           JSR TYPFIL
1885  CE45  F0 03              BEQ CY30
1886  CE47              ;
1887  CE47  20 27 D3           JSR PIBYTE
1888  CE4A              CY30
1889  CE4A  AE 7B 43           LDX F2PTR
1890  CE4D              ;
1891  CE4D  E8                 INX
1892  CE4E  EC 7A 43           CPX F2CNT
1893  CE51  90 D5              BCC CY10        ;MORE FILES TO COPY
1894  CE53  A9 12              LDA #IWSA
1895  CE55  85 16              STA SA
1896  CE57  4C 3A E3           JMP CLSCHN      ;CLOSE COPY CHANNEL, FILE
1897  CE5A              ;
1898  CE5A              OPIRFL
1899  CE5A  AE 7B 43           LDX F2PTR
1900  CE5D  B5 90              LDA FILDRV,X
1901  CE5F  29 01              AND #1
1902  CE61  85 12              STA DRVNUM
1903  CE63  AD 46 FB           LDA DIRTRK
1904  CE66  85 13              STA TRACK
1905  CE68  B5 86              LDA ENTSEC,X
1906  CE6A  85 14              STA SECTOR
1907  CE6C  20 71 D7           JSR OPNIRD
1908  CE6F  AE 7B 43           LDX F2PTR
1909  CE72  B5 8B              LDA ENTIND,X
1910  CE74  20 C6 D7           JSR SETPNT
1911  CE77  AE 7B 43           LDX F2PTR
1912  CE7A  B5 95              LDA PATTYP,X
1913  CE7C  29 07              AND #TYPMSK
1914  CE7E  85 CF              STA TYPE
1915  CE80              ;
1916  CE80  A9 00              LDA #0
1917  CE82  8D 4D 43           STA REC
1918  CE85  20 DD E1           JSR OPREAD
1919  CE88  A0 01              LDY #1
1920  CE8A  20 9C D4           JSR TYPFIL
1921  CE8D  F0 01              BEQ OPIR10
1922  CE8F  C8                 INY
```

```
LINE# LOC   CODE        LINE


1923  CE90              OPIR10
1924  CE90  98                 TYA
1925  CE91  4C C6 D7           JMP SETPNT
1926  CE94              ;
1927  CE94              GIBYTE
1928  CE94  A9 11              LDA #IRSA
1929  CE96  85 16              STA SA
1930  CE98              GCBYTE
1931  CE98  20 9A D6           JSR GBYTE
1932  CE9B              ;
1933  CE9B  85 18              STA DATA
1934  CE9D  A6 15              LDX LINDX
1935  CE9F  B5 A2              LDA CHNRDY,X
1936  CEA1  29 08              AND #EOISND
1937  CEA3  85 AA              STA EOIFLG
1938  CEA5  D0 0A              BNE GIB20
1939  CEA7              ;
1940  CEA7  20 9C D4           JSR TYPFIL
1941  CEAA  F0 05              BEQ GIB20
1942  CEAC              ;
1943  CEAC  A9 80              LDA #LRF
1944  CEAE  20 D8 E5           JSR SETFLG
1945  CEB1              GIB20
1946  CEB1  60                 RTS
1947  CEB2              ;
```

```
LINE# LOC   CODE        LINE


1949  CEB2              ; RENAME FILE NAME IN DIRECTORY
1950  CEB2  20 ED C5    RENAME JSR ALLDRS       ;SET BOTH DRIVE #'S
1951  CEB5  A5 91              LDA FILDRV+1
1952  CEB7  29 01              AND #1
1953  CEB9  85 91              STA FILDRV+1
1954  CEBB  C5 90              CMP FILDRV
1955  CEBD  F0 02              BEQ RN10        ;SAME DRIVE #'S
1956  CEBF  09 80              ORA #$80        ;CHECK BOTH DRIVES FOR NAME
1957  CEC1  85 90       RN10   STA FILDRV
1958  CEC3  20 1B C7           JSR LOOKUP      ;LOOK UP BOTH NAMES
1959  CEC6  20 10 CF           JSR CHKIO       ;CHECK FOR EXISTENCE
1960  CEC9  A5 91              LDA FILDRV+1
1961  CECB  29 01              AND #1
1962  CECD  85 12              STA DRVNUM
1963  CECF  A5 87              LDA ENTSEC+1
1964  CED1  85 14              STA SECTOR
1965  CED3  20 99 E6           JSR RDAB        ;READ DIRECTORY SECTOR
1966  CED6  20 AF D8           JSR WATJOB
1967  CED9  A5 8C              LDA ENTIND+1
1968  CEDB  18                 CLC             ;SET SECTOR INDEX
1969  CEDC  69 03              ADC #3          ;...+3
1970  CEDE  20 C6 D7           JSR SETPNT
1971  CEE1  20 D5 E7           JSR GETACT
1972  CEE4  A8                 TAY
1973  CEE5  AE 7C 43           LDX FILTBL
1974  CEE8  A9 10              LDA #16
1975  CEEA  20 3D C9           JSR TRNAME      ;TRANSFER NAME
1976  CEED  20 A0 E6           JSR WRTOUT      ;WRITE SECTOR OUT
1977  CEF0  20 AF D8           JSR WATJOB
1978  CEF3  4C 66 C4           JMP ENDCMD


1981  CEF6              ; CHECK I/O FILE FOR EXIST
1982  CEF6              CHKIN
1983  CEF6  A5 96              LDA PATTYP+1    ;1ST FILE BEARS TYPE
1984  CEF8  29 07              AND #TYPMSK
1985  CEFA  85 CF              STA TYPE
1986  CEFC              ;
1987  CEFC  AE 7A 43           LDX F2CNT
1988  CEFF  CA          CK10   DEX
1989  CF00  EC 79 43           CPX F1CNT
1990  CF03  90 0A              BCC CK20
1991  CF05  BD 82 43           LDA FILTRK,X
1992  CF08  D0 F5              BNE CK10
1993  CF0A  A9 62              LDA #FLNTFD     ;INPUT FILE NOT FOUND
1994  CF0C  4C 96 C4           JMP CMDERR
1995  CF0F              CK20
1996  CF0F  60                 RTS
1997  CF10              ;
1998  CF10              CHKIO
1999  CF10  20 F6 CE           JSR CHKIN
2000  CF13  BD 82 43    CK25   LDA FILTRK,X
2001  CF16  F0 05              BEQ CK30
2002  CF18  A9 63              LDA #FLEXST
2003  CF1A  4C 96 C4           JMP CMDERR
```

```
LINE# LOC   CODE        LINE


2004  CF1D  CA          CK30    DEX
2005  CF1E  10 F3               BPL CK25
2006  CF20  60                  RTS

2008  CF21                      .END
2009  CF21                      .LIB MEMRW
```

```
LINE# LOC   CODE          LINE

2011  CF21                ; MEMORY ACCESS COMMANDS
2012  CF21                ;   "-" MUST BE 2ND CHAR
2013  CF21  AD 01 43      MEM     LDA CMDBUF+1
2014  CF24  C9 2D                 CMP #'-
2015  CF26  D0 4B                 BNE MEMERR
2016  CF28                ;
2017  CF28  AD 03 43              LDA CMDBUF+3    ;SET ADDRESS IN TEMP
2018  CF2B  85 04                 STA TEMP
2019  CF2D  AD 04 43              LDA CMDBUF+4
2020  CF30  85 05                 STA TEMP+1
2021  CF32                ;
2022  CF32  A0 00                 LDY #0
2023  CF34  AD 02 43              LDA CMDBUF+2
2024  CF37  C9 52                 CMP #'R
2025  CF39  F0 0E                 BEQ MEMRD       ;READ
2026  CF3B  20 39 FB              JSR KILLP       ;KILL PROTECT
2027  CF3E  C9 57                 CMP #'W
2028  CF40  F0 36                 BEQ MEMWRT      ;WRITE
2029  CF42  C9 45                 CMP #'E
2030  CF44  D0 2D                 BNE MEMERR      ;ERROR
2031  CF46                                        ;  EXECUTE
2032  CF46  6C 04 00      MEMEX   JMP (TEMP)

2034  CF49                MEMRD
2035  CF49  B1 04                 LDA (TEMP),Y
2036  CF4B  85 18                 STA DATA
2037  CF4D  AD 76 43              LDA CMDSIZ
2038  CF50  C9 06                 CMP #6
2039  CF52  90 19                 BCC M30
2040  CF54                ;
2041  CF54  AE 05 43              LDX CMDBUF+5
2042  CF57  CA                    DEX
2043  CF58  F0 13                 BEQ M30
2044  CF5A  8A                    TXA
2045  CF5B  18                    CLC
2046  CF5C  65 04                 ADC TEMP
2047  CF5E  E6 04                 INC TEMP
2048  CF60  85 CE                 STA LSTCHR+ERRCHN
2049  CF62  A5 04                 LDA TEMP
2050  CF64  85 47                 STA CB+2
2051  CF66  A5 05                 LDA TEMP+1
2052  CF68  85 48                 STA CB+3
2053  CF6A  4C 40 D7              JMP GE20
2054  CF6D                M30
2055  CF6D  20 64 D4              JSR FNDRCH
2056  CF70  4C 37 D7              JMP GE15

2058  CF73  A9 31         MEMERR LDA #BADCMD       ;BAD COMMAND
2059  CF75  4C 96 C4              JMP CMDERR

2061  CF78                MEMWRT                   ;WRITE
2062  CF78  B9 06 43      M10     LDA CMDBUF+6,Y
2063  CF7B  91 04                 STA (TEMP),Y    ;TRANSFER FROM CMDBUF
2064  CF7D  C8                    INY
2065  CF7E  CC 05 43              CPY CMDBUF+5    ;# OF BYTES TO WRITE
```

```
LINE# LOC    CODE         LINE


2066  CF81  90 F5              BCC M10
2067  CF83  60                 RTS

2069  CF84                     .END
2070  CF84                     .LIB BLOCK
```

```
LINE# LOC   CODE         LINE


2072  CF84               ;  ROM 1.1 ADDITIONS
2073  CF84               ; USER COMMANDS

2075  CF84  AC 01 43     USER    LDY CMDBUF+1
2076  CF87  C0 30                CPY #'0
2077  CF89  D0 09                BNE US10        ;0 RESETS PNTR

2079  CF8B  A9 EA        USRINT LDA #<UBLOCK     ;SET DEFAULT BLOCK ADD
2080  CF8D  85 00                STA USRJMP
2081  CF8F  A9 FF                LDA #>UBLOCK
2082  CF91  85 01                STA USRJMP+1
2083  CF93  60                   RTS

2085  CF94  20 9A CF     US10    JSR USREXC      ;EXECUTE CODE BY TABLE
2086  CF97  4C 66 C4             JMP ENDCMD

2088  CF9A  88           USREXC DEY              ;ENTRY IS(((INDEX-1)AND$F)
2089  CF9B  98                   TYA
2090  CF9C  29 0F                AND #$F
2091  CF9E  0A                   ASL A
2092  CF9F  A8                   TAY
2093  CFA0  B1 00                LDA (USRJMP)Y
2094  CFA2  85 0A                STA IP
2095  CFA4  C8                   INY
2096  CFA5  B1 00                LDA (USRJMP)Y
2097  CFA7  85 0B                STA IP+1
2098  CFA9  6C 0A 00             JMP (IP)
```

```
LINE# LOC   CODE          LINE


2100  CFAC                ; OPEN DIRECT ACCESS BUFFER
2101  CFAC                ;  FROM OPEN "#"

2103  CFAC                OPNBLK
2104  CFAC  AD 90 43             LDA LSTDRV
2105  CFAF  85 12                STA DRVNUM
2106  CFB1  AE 76 43             LDX CMDSIZ
2107  CFB4  CA                   DEX
2108  CFB5  D0 0D                BNE OB10

2110  CFB7  A9 01                LDA #1         ;GET ANY BUFFER
2111  CFB9  20 59 D5             JSR GETRCH
2112  CFBC  4C 0E D0             JMP OB30

2114  CFBF  A9 70         OB05   LDA #NOCHNL
2115  CFC1  4C 96 C4             JMP CMDERR

2117  CFC4  A0 01         OB10   LDY #1          ;BUFFER # IS REQUESTED
2118  CFC6  20 97 D0             JSR BP05
2119  CFC9  AE 87 43             LDX FILSEC
2120  CFCC  E0 0C                CPX #BAMJOB    ;MUST BE LESS THAN 13.
2121  CFCE  B0 EF                BCS OB05

2123  CFD0  A9 00                LDA #0
2124  CFD2  85 04                STA TEMP
2125  CFD4  85 05                STA TEMP+1
2126  CFD6  38                   SEC

2128  CFD7                OB15
2129  CFD7  26 04                ROL TEMP
2130  CFD9  26 05                ROL TEMP+1
2131  CFDB  CA                   DEX
2132  CFDC  10 F9                BPL OB15

2134  CFDE  A5 04                LDA TEMP
2135  CFE0  2D 40 43             AND BUFUSE
2136  CFE3  D0 DA                BNE OB05       ;BUFFER IS USED
2137  CFE5  A5 05                LDA TEMP+1
2138  CFE7  2D 41 43             AND BUFUSE+1
2139  CFEA  D0 D3                BNE OB05       ;BUF IS USED

2141  CFEC  A5 04                LDA TEMP
2142  CFEE  0D 40 43             ORA BUFUSE     ;SET BUFFER AS USED
2143  CFF1  8D 40 43             STA BUFUSE
2144  CFF4  A5 05                LDA TEMP+1
2145  CFF6  0D 41 43             ORA BUFUSE+1
2146  CFF9  8D 41 43             STA BUFUSE+1

2148  CFFC  A9 00                LDA #0         ;SET UP CHANNEL
2149  CFFE  20 59 D5             JSR GETRCH
2150  D001  A6 15                LDX LINDX
2151  D003  AD 87 43             LDA FILSEC
2152  D006  95 49                STA BUF0,X
2153  D008  AA                   TAX
2154  D009  A5 12                LDA DRVNUM
```

```
LINE# LOC   CODE          LINE

2155  D00B  9D 50 43                STA LSTJOB,X

2157  D00E  A6 16          0B30     LDX SA
2158  D010  B5 AC                   LDA LINTAB,X    ;SET LINDX TABLE
2159  D012  09 40                   ORA #$40
2160  D014  95 AC                   STA LINTAB,X

2162  D016  A4 15                   LDY LINDX
2163  D018  A9 FF                   LDA #$FF
2164  D01A  99 C7 00                STA LSTCHR,Y

2166  D01D  A9 89                   LDA #RNDRDY
2167  D01F  99 A2 00                STA CHNRDY,Y    ;SET CHANNEL READY

2169  D022  B9 49 00                LDA BUF0,Y
2170  D025  99 BF 00                STA CHNDAT,Y    ;BUFFER # AS 1ST CHAR
2171  D028  0A                      ASL A
2172  D029  AA                      TAX
2173  D02A  A9 01                   LDA #1
2174  D02C  95 29                   STA BUFTAB,X
2175  D02E  A9 0E                   LDA #DIRTYP+DIRTYP
2176  D030  99 9A 00                STA FILTYP,Y    ;SET DIRECT FILE TYPE
2177  D033  4C 66 C4                JMP ENDCMD
```

```
LINE# LOC   CODE        LINE


2179  D036              ; BLOCK COMMANDS
2180  D036  A0 00       BLOCK   LDY #0
2181  D038  A2 00               LDX #0
2182  D03A  A9 2D               LDA #'-        ;"-" SEPARATES CMD FROM SUBCMD
2183  D03C  20 36 C5            JSR PARSE      ;LOCATE SUB-CMD
2184  D03F  D0 0A               BNE BLK40

2186  D041  A9 31       BLK10   LDA #BADCMD
2187  D043  4C 96 C4            JMP CMDERR

2189  D046  A9 30       BLK30   LDA #BADSYN
2190  D048  4C 96 C4            JMP CMDERR

2192  D04B  8A          BLK40   TXA
2193  D04C  D0 F8               BNE BLK30

2195  D04E  A2 05               LDX #NBCMDS-1  ;FIND COMMAND
2196  D050  B9 00 43            LDA CMDBUF,Y
2197  D053  DD 78 D0    BLK50   CMP BCTAB,X
2198  D056  F0 05               BEQ BLK60
2199  D058  CA                  DEX
2200  D059  10 F8               BPL BLK50
2201  D05B  30 E4               BMI BLK10

2203  D05D              BLK60
2204  D05D  8A                  TXA
2205  D05E  09 80               ORA #$80
2206  D060  8D 3B 43            STA CMDNUM
2207  D063  20 8A D0            JSR BLKPAR     ;PARSE PARMS

2209  D066  AD 3B 43            LDA CMDNUM
2210  D069  0A                  ASL A
2211  D06A  AA                  TAX
2212  D06B  BD 7F D0            LDA BCJMP+1,X
2213  D06E  85 05               STA TEMP+1
2214  D070  BD 7E D0            LDA BCJMP,X
2215  D073  85 04               STA TEMP

2217  D075  6C 04 00            JMP (TEMP)     ;GOTO COMMAND

2219  D078  41 46       BCTAB   .BYT 'AFRWEP'
2220  D07E              NBCMDS =*-BCTAB

2222  D07E  19 D1       BCJMP   .WORD BLKALC    ;BLOCK-ALLOCATE
2223  D080  10 D1               .WORD BLKFRE   ;BLOCK-FREE
2224  D082  6C D1               .WORD BLKRD    ;BLOCK-READ
2225  D084  89 D1               .WORD BLKWT    ;BLOCK-WRITE
2226  D086  BB D1               .WORD BLKEXC   ;BLOCK-EXECUTE
2227  D088  D5 D1               .WORD BLKPTR   ;BLOCK-POINTER

2229  D08A  A0 00       BLKPAR LDY #0           ;PARSE BLOCK PARMS
2230  D08C  A2 00               LDX #0
2231  D08E  A9 3A               LDA #':
2232  D090  20 36 C5            JSR PARSE
2233  D093  D0 02               BNE BP05       ;FOUND ":"
```

```
LINE# LOC   CODE          LINE


2235  D095  A0 03                LDY #3          ;ELSE CHAR #3 IS BEGINNING
2236  D097  B9 00 43      BP05   LDA CMDBUF,Y
2237  D09A  C9 20                CMP #'
2238  D09C  F0 08                BEQ BP10

2240  D09E  C9 1D                CMP #29         ;SKIP CHARACTER
2241  D0A0  F0 04                BEQ BP10

2243  D0A2  C9 2C                CMP #',
2244  D0A4  D0 07                BNE BP20

2246  D0A6  C8            BP10   INY
2247  D0A7  CC 76 43             CPY CMDSIZ
2248  D0AA  90 EB                BCC BP05

2250  D0AC  60                   RTS             ;THAT'S ALL

2252  D0AD  20 BC D0      BP20   JSR ASCHEX
2253  D0B0  EE 79 43             INC F1CNT
2254  D0B3  AC 7B 43             LDY F2PTR
2255  D0B6  E0 04                CPX #MXFILS-1
2256  D0B8  90 EC                BCC BP10

2258  D0BA  B0 8A                BCS BLK30       ;BAD SYNTAX

2260  D0BC                ; CONVERT ASCII TO HEX (BINARY)
2261  D0BC                ;  & STORE CONVERSION IN TABLES
2262  D0BC                ;   .Y= PTR INTO CMDBUF
2263  D0BC  A9 00         ASCHEX LDA #0
2264  D0BE  85 04                STA TEMP
2265  D0C0  85 05                STA TEMP+1
2266  D0C2  85 07                STA TEMP+3

2268  D0C4  A2 FF                LDX #$FF
2269  D0C6  B9 00 43      AH10   LDA CMDBUF,Y     ;TEST FOR DEC #
2270  D0C9  C9 40                CMP #$40
2271  D0CB  B0 18                BCS AH20        ;NON-NUMERIC TERMINATES
2272  D0CD  C9 30                CMP #$30
2273  D0CF  90 14                BCC AH20        ;NON-NUMERIC

2275  D0D1  29 0F                AND #$F
2276  D0D3  48                   PHA
2277  D0D4  A5 05                LDA TEMP+1      ;SHIFT DIGITS (*10)
2278  D0D6  85 06                STA TEMP+2
2279  D0D8  A5 04                LDA TEMP
2280  D0DA  85 05                STA TEMP+1
2281  D0DC  68                   PLA
2282  D0DD  85 04                STA TEMP
2283  D0DF  C8                   INY
2284  D0E0  CC 76 43             CPY CMDSIZ
2285  D0E3  90 E1                BCC AH10        ;STILL IN STRING

2287  D0E5  8C 7B 43      AH20   STY F2PTR        ;CONVERT DIGITS TO...
2288  D0E8  18                   CLC             ;...BINARY BY DEC TABLE
2289  D0E9  A9 00                LDA #0
```

```
LINE# LOC   CODE        LINE

2291  D0EB  E8          AH30    INX
2292  D0EC  E0 03               CPX #3
2293  D0EE  B0 0F               BCS AH40

2295  D0F0  B4 04               LDY TEMP,X
2296  D0F2  88          AH35    DEY
2297  D0F3  30 F6               BMI AH30

2299  D0F5  7D 0D D1            ADC DECTAB,X
2300  D0F8  90 F8               BCC AH35

2302  D0FA  18                  CLC
2303  D0FB  E6 07               INC TEMP+3
2304  D0FD  D0 F3               BNE AH35

2306  D0FF  48          AH40    PHA
2307  D100  AE 79 43            LDX F1CNT
2308  D103  A5 07               LDA TEMP+3
2309  D105  9D 82 43            STA FILTRK,X    ;STORE RESULT IN TABLE
2310  D108  68                  PLA
2311  D109  9D 87 43            STA FILSEC,X
2312  D10C  60                  RTS

2314  D10D  01          DECTAB .BYT 1,10,100    ;DECIMAL TABLE
2314  D10E  0A
2314  D10F  64

2316  D110              ;BLOCK-FREE
2317  D110  20 0D D2    BLKFRE JSR BLKTST
2318  D113  20 77 F8            JSR FRETS
2319  D116  4C 66 C4            JMP ENDCMD

2321  D119              ;BLOCK-ALLOCATE
2322  D119  20 0D D2    BLKALC JSR BLKTST

2324  D11C              BA10
2325  D11C  A5 14               LDA SECTOR
2326  D11E  48                  PHA
2327  D11F  20 D4 FA            JSR GETSEC
2328  D122  F0 0B               BEQ BA15        ;NONE GREATER ON THIS TRACK
2329  D124  68                  PLA
2330  D125  C5 14               CMP SECTOR
2331  D127  D0 19               BNE BA30        ;REQUESTED SECTOR NOT AVAIL
2332  D129  20 AB F8            JSR USEDTS
2333  D12C  4C 66 C4            JMP ENDCMD
2334  D12F              ;
2335  D12F              BA15
2336  D12F  68                  PLA             ;POP STACK
2337  D130              BA20
2338  D130  A9 00               LDA #0
2339  D132  85 14               STA SECTOR
2340  D134  E6 13               INC TRACK
2341  D136  A5 13               LDA TRACK
2342  D138  CD B0 10            CMP MAXTRK
2343  D13B  B0 0A               BCS BA40        ;GONE ALL THE WAY
```

```
LINE# LOC   CODE        LINE


2344  D13D              ;
2345  D13D  20 D4 FA            JSR GETSEC
2346  D140  F0 EE               BEQ BA20
2347  D142              BA30
2348  D142  A9 65               LDA #NOBLK
2349  D144  20 EA EE            JSR CMDER2
2350  D147              BA40
2351  D147  A9 65               LDA #NOBLK
2352  D149  20 96 C4            JSR CMDERR      ;T=0,S=0 :NONE LEFT
2353  D14C              ;

2355  D14C              ; BLOCK READ SUBS
2356  D14C  20 0A D2    BLKRD2 JSR BKOTST      ;TEST PARMS
2357  D14F  4C 5C D7            JMP DRTRD

2359  D152  20 A6 D4    GETSIM JSR GETPRE      ;GET BYTE W/O INC
2360  D155  A1 29               LDA (BUFTAB,X)
2361  D157  60                  RTS

2363  D158              ; BLOCK READ
2364  D158  20 4C D1    BLKRD3 JSR BLKRD2
2365  D15B  A9 00               LDA #0
2366  D15D  20 C6 D7            JSR SETPNT
2367  D160  20 52 D1            JSR GETSIM      ;Y=LINDX
2368  D163

2370  D163  99 C7 00            STA LSTCHR,Y
2371  D166  A9 89               LDA #RNDRDY
2372  D168  99 A2 00            STA CHNRDY,Y
2373  D16B  60                  RTS
2374  D16C              BLKRD
2375  D16C  20 58 D1            JSR BLKRD3
2376  D16F  20 EA D6            JSR RNGET1
2377  D172  4C 66 C4            JMP ENDCMD

2379  D175              ;USER DIRECT READ, LSTCHR=$FF
2380  D175              UBLKRD
2381  D175  20 8A D0            JSR BLKPAR
2382  D178  20 58 D1            JSR BLKRD3
2383  D17B  B9 C7 00            LDA LSTCHR,Y
2384  D17E  99 BF 00            STA CHNDAT,Y
2385  D181  A9 FF               LDA #$FF
2386  D183  99 C7 00            STA LSTCHR,Y
2387  D186  4C 66 C4            JMP ENDCMD      ;(RTS)


2389  D189              ;BLOCK-WRITE
2390  D189  20 0A D2    BLKWT  JSR BKOTST

2392  D18C  20 E6 D7            JSR GETPNT
2393  D18F  A8                  TAY
2394  D190  88                  DEY
2395  D191  C9 02               CMP #2
2396  D193  B0 02               BCS BW10
2397  D195  A0 01               LDY #1
```

```
LINE# LOC   CODE        LINE


2399  D197  A9 00       BW10   LDA #0          ;SET RECORD SIZE
2400  D199  20 C6 D7           JSR SETPNT
2401  D19C  98                 TYA
2402  D19D  20 73 D3           JSR PUTBYT
2403  D1A0  8A                 TXA
2404  D1A1  48                 PHA

2406  D1A2  20 60 D7    BW20   JSR DRTWRT      ;WRITE BLOCK
2407  D1A5  68                 PLA
2408  D1A6  AA                 TAX
2409  D1A7  A4 15              LDY LINDX
2410  D1A9  20 EC D6           JSR RNGET2
2411  D1AC  4C 66 C4           JMP ENDCMD

2413  D1AF              ;USER DIRCT WRITE, NO LSTCHR
2414  D1AF  20 8A D0    UBLKWT JSR BLKPAR
2415  D1B2  20 0A D2           JSR BKOTST
2416  D1B5  20 60 D7           JSR DRTWRT
2417  D1B8  4C 66 C4           JMP ENDCMD

2419  D1BB              ;IN .FILE VECTOR:
2420  D1BB              ;*=$FFFA-6 ;USER DIRECT ACCESS
2421  D1BB              ;UBLOCK .WORD UBLKRD
2422  D1BB              ;       .WORD UBLKWT


2424  D1BB              ;BLOCK-EXECUTE
2425  D1BB              BLKEXC
2426  D1BB  20 39 FB           JSR KILLP      ;KILL PROTECT
2427  D1BE  20 4C D1           JSR BLKRD2     ;READ BLOCK & EXECUTE
2428  D1C1  A9 00              LDA #0

2430  D1C3  85 04       BE05   STA TEMP
2431  D1C5  A6 AB              LDX JOBNUM
2432  D1C7  BD 04 D8           LDA BUFIND,X
2433  D1CA  85 05              STA TEMP+1
2434  D1CC  20 D2 D1           JSR BE10       ;INDIRECT JSR
2435  D1CF  4C 66 C4           JMP ENDCMD

2437  D1D2  6C 04 00    BE10   JMP (TEMP)


2439  D1D5              ;BUFFER-POINTER, SET BUFFER POINTER
2440  D1D5  20 EA D1    BLKPTR JSR BUFTST
2441  D1D8  A5 AB              LDA JOBNUM
2442  D1DA  0A                 ASL A
2443  D1DB  AA                 TAX
2444  D1DC  AD 88 43           LDA FILSEC+1
2445  D1DF  95 29              STA BUFTAB,X
2446  D1E1  20 A6 D4           JSR GETPRE
2447  D1E4  20 EC D6           JSR RNGET2     ;SET UP GET
2448  D1E7  4C 66 C4           JMP ENDCMD

2450  D1EA              ;TEST FOR ALLOCATED BUFFER..
2451  D1EA              ;  ..RELATED TO SA
```

```
LINE# LOC   CODE        LINE


2452  D1EA  A6 81       BUFTST LDX F1PTR
2453  D1EC  E6 81              INC F1PTR
2454  D1EE  BD 87 43           LDA FILSEC,X
2455  D1F1  A8                 TAY
2456  D1F2  88                 DEY
2457  D1F3  88                 DEY
2458  D1F4  C0 0C              CPY #BAMJOB
2459  D1F6  90 05              BCC BT20

2461  D1F8  A9 70       BT15   LDA #NOCHNL
2462  D1FA  4C 96 C4           JMP CMDERR

2464  D1FD  85 16       BT20   STA SA
2465  D1FF  20 64 D4           JSR FNDRCH
2466  D202  B0 F4              BCS BT15
2467  D204  20 D5 E7           JSR GETACT
2468  D207  85 AB              STA JOBNUM
2469  D209  60                 RTS

2471  D20A                     ;TEST BLOCK OPERATION PARMS
2472  D20A  20 EA D1    BKOTST JSR BUFTST
2473  D20D                     ;
2474  D20D                     ;TEST FOR LEGAL BLOCK &..
2475  D20D                     ;   ..SET UP DRV, TRK, SEC
2476  D20D  A6 81       BLKTST LDX F1PTR
2477  D20F  BD 87 43           LDA FILSEC,X
2478  D212  29 01              AND #1
2479  D214  85 12              STA DRVNUM
2480  D216  BD 89 43           LDA FILSEC+2,X
2481  D219  85 14              STA SECTOR
2482  D21B  BD 88 43           LDA FILSEC+1,X
2483  D21E  85 13              STA TRACK
2484  D220                BT05
2485  D220  20 74 D8           JSR TSCHK
2486  D223  4C DC F9           JMP SETLDS      ;(RTS)

2488  D226                     .END
2489  D226                     .LIB FNDREL
```

```
LINE# LOC   CODE         LINE


2491  D226               ;********************************
2492  D226               ;*
2493  D226               ;* FIND RELATIVE FILE
2494  D226               ;*
2495  D226               ;* VERSION 2.5
2496  D226               ;*
2497  D226               ;*
2498  D226               ;* INPUTS
2499  D226               ;*  RECL    - 1BYTE=LO RECORD #
2500  D226               ;*  RECH    - 1BYTE=HI RECORD #
2501  D226               ;*  RS - 1BYTE=RECORD SIZE
2502  D226               ;*  RECPTR - 1BYTE=FIRST BYTE
2503  D226               ;*            WANTED FROM RECORD
2504  D226               ;*
2505  D226               ;* OUTPUTS
2506  D226               ;*  SSNUM  - 1BYTE=SIDE SECTOR #
2507  D226               ;*  SSIND  - 1BYTE=INDEX INTO SS
2508  D226               ;*  RELPTR - 1BYTE=PTR TO FIRST
2509  D226               ;*            BYTE WANTED
2510  D226               ;*
2511  D226               ;********************************
2512  D226               ;
2513  D226               ;
2514  D226               ;
2515  D226               ;
2516  D226               ;
2517  D226               ;
2518  D226  20 44 D2     FNDREL JSR MULPLY      ;RESULT=RN*RS+RP
2519  D229  20 86 D2            JSR DIV254     ;DIVIDE BY 254
2520  D22C  A5 23               LDA ACCUM+1    ;SAVE REMAINDER
2521  D22E  85 85               STA RELPTR
2522  D230  20 89 D2            JSR DIV120     ;DIVIDE BY 120
2523  D233  E6 85               INC RELPTR
2524  D235  E6 85               INC RELPTR
2525  D237  A5 1E               LDA RESULT     ;SAVE QUOTIENT
2526  D239  85 83               STA SSNUM
2527  D23B  A5 23               LDA ACCUM+1    ;SAVE REMAINDER
2528  D23D  0A                  ASL A          ;CALC INDEX INTO SS
2529  D23E  18                  CLC
2530  D23F  69 10               ADC #16        ;SKIP LINK TABLE
2531  D241  85 84               STA SSIND
2532  D243  60                  RTS
2533  D244               ;
2534  D244               ;
2535  D244               ;
2536  D244               ;  MULTIPLY
2537  D244               ;
2538  D244               ;  RESULT=RECNUM*RS+RECPTR
2539  D244               ;
2540  D244               ;   DESTROYS A,X
2541  D244               ;
2542  D244  20 F1 D2     MULPLY JSR ZERRES      ;RESULT=0
2543  D247  85 25               STA ACCUM+3    ;A=0
2544  D249  A6 15               LDX LINDX      ;GET INDEX
2545  D24B  B5 59               LDA RECL,X
```

```
LINE# LOC   CODE        LINE

2546  D24D  85 23                STA ACCUM+1
2547  D24F  B5 61                LDA RECH,X
2548  D251  85 24                STA ACCUM+2
2549  D253  D0 04                BNE MUL25        ;ADJUST FOR REC #1 &...
2550  D255  A5 23                LDA ACCUM+1      ;...#0 = 1ST REC
2551  D257  F0 0B                BEQ MUL50
2552  D259  A5 23         MUL25  LDA ACCUM+1
2553  D25B  38                   SEC
2554  D25C  E9 01                SBC #1
2555  D25E  85 23                STA ACCUM+1
2556  D260  B0 02                BCS MUL50
2557  D262  C6 24                DEC ACCUM+2
2558  D264                MUL50
2559  D264  B5 71                LDA RS,X         ;COPY RECSIZ
2560  D266  85 04                STA TEMP
2561  D268  46 04         MUL100 LSR TEMP          ;DO AN ADD ?
2562  D26A  90 03                BCC MUL200       ;NO
2563  D26C  20 05 D3             JSR ADDRES       ;RESULT=RESULT+ACCUM+1,2,3
2564  D26F  20 FD D2      MUL200 JSR ACCX2         ;2*(ACCUM+1,2,3)
2565  D272  A5 04                LDA TEMP         ;DONE ?
2566  D274  D0 F2                BNE MUL100       ;NO
2567  D276  A5 82                LDA RECPTR       ;ADD IN LAST BIT
2568  D278  18                   CLC
2569  D279  65 1E                ADC RESULT
2570  D27B  85 1E                STA RESULT
2571  D27D  90 06                BCC MUL400       ;SKIP NO CARRY
2572  D27F  E6 1F                INC RESULT+1
2573  D281  D0 02                BNE MUL400
2574  D283  E6 20                INC RESULT+2
2575  D285  60            MUL400 RTS
2576  D286                ;
2577  D286                ;
2578  D286                ;
2579  D286                ; DIVIDE
2580  D286                ;
2581  D286                ; RESULT=QUOTIENT ,ACCUM+1=REMAINDER
2582  D286                ;
2583  D286                ;  DESTROYS A,X
2584  D286                ;
2585  D286  A9 FE         DIV254 LDA #254          ;DIVIDE BY 254
2586  D288  2C                   .BYTE $2C        ;SKIP TWO BYTES
2587  D289  A9 78         DIV120 LDA #120          ;DIVIDE BY 120
2588  D28B  85 04                STA TEMP         ;SAVE DIVISOR
2589  D28D  A2 03                LDX #3           ;SWAP ACCUM+1,2,3 WITH
2590  D28F  B5 22         DIV100 LDA ACCUM,X        ;  RESULT,1,2
2591  D291  48                   PHA
2592  D292  B5 1D                LDA RESULT-1,X
2593  D294  95 22                STA ACCUM,X
2594  D296  68                   PLA
2595  D297  95 1D                STA RESULT-1,X
2596  D299  CA                   DEX
2597  D29A  D0 F3                BNE DIV100
2598  D29C  20 F1 D2             JSR ZERRES       ;RESULT=0
2599  D29F  A2 00         DIV150 LDX #0
2600  D2A1  B5 23         DIV200 LDA ACCUM+1,X     ;DIVIDE BY 256
```

```
LINE# LOC   CODE         LINE

2601  D2A3  95 22               STA ACCUM,X
2602  D2A5  E8                  INX
2603  D2A6  E0 04               CPX #4          ;DONE ?
2604  D2A8  90 F7               BCC DIV200      ;NO
2605  D2AA  A9 00               LDA #0          ;ZERO HI BYTE
2606  D2AC  85 25               STA ACCUM+3
2607  D2AE  24 04               BIT TEMP        ;A DIV120 ?
2608  D2B0  30 09               BMI DIV300      ;NO
2609  D2B2  06 22               ASL ACCUM       ;ONLY DIVIDE BY 128
2610  D2B4  08                  PHP             ;SAVE CARRY
2611  D2B5  46 22               LSR ACCUM       ;NORMALIZE
2612  D2B7  28                  PLP             ;RESTORE CARRY
2613  D2B8  20 FE D2            JSR ACC200      ;2*(X/256)=X/128
2614  D2BB  20 05 D3     DIV300 JSR ADDRES       ;TOTAL A QUOTIENT
2615  D2BE  20 FD D2            JSR ACCX2       ;A=2*A
2616  D2C1  24 04               BIT TEMP        ;A DIV120 ?
2617  D2C3  30 03               BMI DIV400      ;NO
2618  D2C5  20 FA D2            JSR ACCX4       ;A=4*(2*A)=8*A
2619  D2C8  A5 22        DIV400 LDA ACCUM        ;ADD IN REMAINDER
2620  D2CA  18                  CLC
2621  D2CB  65 23               ADC ACCUM+1
2622  D2CD  85 23               STA ACCUM+1
2623  D2CF  90 06               BCC DIV500
2624  D2D1  E6 24               INC ACCUM+2
2625  D2D3  D0 02               BNE DIV500
2626  D2D5  E6 25               INC ACCUM+3
2627  D2D7  A5 25        DIV500 LDA ACCUM+3      ;TEST < 256
2628  D2D9  05 24               ORA ACCUM+2
2629  D2DB  D0 C2               BNE DIV150      ;CRUNCH SOME MORE
2630  D2DD  A5 23               LDA ACCUM+1     ;IS REMAINDER < DIVISOR
2631  D2DF  38                  SEC
2632  D2E0  E5 04               SBC TEMP
2633  D2E2  90 0C               BCC DIV700      ;YES
2634  D2E4  E6 1E               INC RESULT      ;NO - FIX RESULT
2635  D2E6  D0 06               BNE DIV600
2636  D2E8  E6 1F               INC RESULT+1
2637  D2EA  D0 02               BNE DIV600
2638  D2EC  E6 20               INC RESULT+2
2639  D2EE  85 23        DIV600 STA ACCUM+1      ;NEW REMAINDER
2640  D2F0  60           DIV700 RTS
2641  D2F1               ;
2642  D2F1               ;
2643  D2F1               ;
2644  D2F1               ;  ZERO RESULT
2645  D2F1               ;
2646  D2F1  A9 00        ZERRES LDA #0
2647  D2F3  85 1E               STA RESULT
2648  D2F5  85 1F               STA RESULT+1
2649  D2F7  85 20               STA RESULT+2
2650  D2F9  60                  RTS
2651  D2FA               ;
2652  D2FA               ;
2653  D2FA               ;
2654  D2FA               ;  MULTIPLY ACCUM BY 4
2655  D2FA               ;
```

```
LINE# LOC   CODE        LINE

2656  D2FA  20 FD D2    ACCX4  JSR ACCX2
2657  D2FD              ;
2658  D2FD              ; MULTIPLY ACCUM BY 2
2659  D2FD              ;
2660  D2FD  18          ACCX2  CLC
2661  D2FE  26 23       ACC200 ROL ACCUM+1
2662  D300  26 24              ROL ACCUM+2
2663  D302  26 25              ROL ACCUM+3
2664  D304  60                 RTS
2665  D305              ;
2666  D305              ;
2667  D305              ;
2668  D305              ; ADD ACCUM TO RESULT
2669  D305              ;
2670  D305              ;  RESULT=RESULT+ACCUM+1,2,3
2671  D305              ;
2672  D305  18          ADDRES CLC
2673  D306  A2 FD              LDX #$FD
2674  D308  B5 21       ADD100 LDA RESULT+3,X
2675  D30A  75 26              ADC ACCUM+4,X
2676  D30C  95 21              STA RESULT+3,X
2677  D30E  E8                 INX
2678  D30F  D0 F7              BNE ADD100
2679  D311  60                 RTS
2680  D312              ;
2681  D312              ;
2682  D312              ;
2683  D312              ;
2684  D312                     .END
2685  D312                     .LIB TST2
```

```
LINE# LOC   CODE        LINE

2687  D312              ;
2688  D312              ; TOGGLE ACTIVE BUFFER # IN BUFNUM
2689  D312              ;
2690  D312  A6 15       DBLBUF LDX LINDX
2691  D314  B5 49              LDA BUFO,X      ;TOGGLE ACTIVE FLAG
2692  D316  49 80              EOR #$80
2693  D318  95 49              STA BUFO,X
2694  D31A  B5 51       DBL1   LDA BUF1,X       ;TOGGLE ACTIVE FLAG
2695  D31C  49 80              EOR #$80
2696  D31E  95 51              STA BUF1,X
2697  D320  20 D5 E7    DBL2   JSR GETACT       ;GET ACTIVE BUF#
2698  D323  AA                 TAX
2699  D324  4C AF D8           JMP WATJOB      ; AND WAIT FOR BUFFER READY
2700  D327              ;
2701  D327              PIBYTE
2702  D327  A2 12              LDX #IWSA
2703  D329  86 16              STX SA
2704  D32B              PBYTE
2705  D32B  20 7F D4           JSR FNDWCH
2706  D32E  20 DC F9           JSR SETLDS
2707  D331  A5 16              LDA SA
2708  D333  C9 0F              CMP #15
2709  D335  F0 23              BEQ L42
2710  D337  D0 08              BNE L40
2711  D339              ;
2712  D339              ; MAIN ROUTINE TO WRITE TO CHANL
2713  D339              ;
2714  D339  A5 17       PUT    LDA ORGSA        ;IS CHANL CMD OR DATA
2715  D33B  29 8F              AND #$8F
2716  D33D  C9 0F              CMP #15         ;<15
2717  D33F  B0 19              BCS L42
2718  D341  20 9C D4    L40    JSR TYPFIL       ;DATA BYTE TO STORE
2719  D344  B0 05              BCS L41         ;BRANCH IF RND
2720  D346  A5 18              LDA DATA        ;SEQ FILE
2721  D348  4C 14 D5           JMP WRTBYT      ;WRITE BYTE TO CHANL
2722  D34B              L41
2723  D34B  D0 03              BNE L46
2724  D34D  4C D4 E8           JMP WRTREL
2725  D350  A5 18       L46    LDA DATA         ;RND FILE WRITE
2726  D352  20 73 D3           JSR PUTBYT      ;WRITE TO CHANL
2727  D355  A4 15              LDY LINDX       ; PREPARE NXT BYTE
2728  D357  4C EC D6           JMP RNGET2
2729  D35A  A9 06       L42    LDA #CMDCHN      ;WRITE TO CMD CHANL
2730  D35C  85 15              STA LINDX
2731  D35E  20 E6 D7           JSR GETPNT      ;TEST IF COMM AND BUFFER FULL
2732  D361  C9 3B              CMP #<CMDBUF+CMDLEN+1
2733  D363  F0 05              BEQ L50         ;IT IS FULL (>40)
2734  D365  A5 18              LDA DATA        ;NOT FULL YET
2735  D367  20 73 D3           JSR PUTBYT      ;STORE THE BYTE
2736  D36A  A5 AA       L50    LDA EOIFLG       ;TST IF LST BYTE OF MSG
2737  D36C  F0 01              BEQ L45         ;IT IS
2738  D36E  60                 RTS             ;NOT YET , RETURN
2739  D36F              L45
2740  D36F  EE 4A 43           INC CMDWAT      ;SET CMD WAITING FLAG
2741  D372  60                 RTS
```

```
LINE# LOC   CODE       LINE

2742  D373              ;
2743  D373              ; PUT .A INTO ACTIVE BUFFER OF LINDX
2744  D373              ;
2745  D373  48          PUTBYT PHA              ; SAVE .A
2746  D374  20 D5 E7           JSR GETACT      ;GET ACTIVE BUF#
2747  D377  10 06              BPL PUTB1       ;BRACH IF THERE IS ONE
2748  D379  68                 PLA             ;NO BUFFER ERROR
2749  D37A  A9 61              LDA #FILNOP
2750  D37C  4C 96 C4           JMP CMDERR      ; JMP TO ERROR ROUTINE
2751  D37F  0A          PUTB1  ASL A            ;SAVE THE BYTE IN BUFFER
2752  D380  AA                 TAX
2753  D381  68                 PLA
2754  D382  81 29              STA (BUFTAB,X)
2755  D384  F6 29              INC BUFTAB,X    ; INC THE BUFFER POINTER
2756  D386  60                 RTS             ;Z=1 IF LAST CHAR SLOT IN BUFF
2757  D387              ;
2758  D387              ; FIND THE ACTIVE BUFFER # (LINDX)
2759  D387              ;
2760  D387              ;
2761  D387              ;INITIALIZE DRIVES (COMMAND)
2762  D387  20 9F C4    INTDRV JSR SIMPRS
2763  D38A  20 B2 D3           JSR INITDR
2764  D38D  AD 8D 43           LDA IMAGE       ;USED AS FLAG FOR BOTH DRIVES
2765  D390  10 0C              BPL ID20
2766  D392  20 5B C6           JSR TOGDRV
2767  D395  20 DC F9           JSR SETLDS
2768  D398  20 B2 D3           JSR INITDR
2769  D39B  20 5B C6           JSR TOGDRV
2770  D39E  4C 66 C4    ID20   JMP ENDCMD

2772  D3A1              ; INITIALIZE DRIVE (DRVNUM)
2773  D3A1              ;
2774  D3A1              ITRIAL
2775  D3A1  A9 0E              LDA #BAMJOB+2
2776  D3A3  85 AB              STA JOBNUM
2777  D3A5  AE 46 FB           LDX DIRTRK
2778  D3A8  86 13              STX TRACK
2779  D3AA  20 DC D9           JSR SETH
2780  D3AD  A9 B0              LDA #SEEK
2781  D3AF  4C A2 D8           JMP DOJOB
2782  D3B2              ;
2783  D3B2  20 59 D6    INITDR JSR CLDCHN
2784  D3B5  20 A1 D3           JSR ITRIAL
2785  D3B8  A6 12              LDX DRVNUM
2786  D3BA  A9 00              LDA #0
2787  D3BC  9D 46 43           STA MDIRTY,X
2788  D3BF  8A                 TXA
2789  D3C0  0A                 ASL A
2790  D3C1  AA                 TAX
2791  D3C2  B9 21 10           LDA HDRS,Y
2792  D3C5  9D 42 43           STA DSKID,X
2793  D3C8  B9 22 10           LDA HDRS+1,Y
2794  D3CB  9D 43 43           STA DSKID+1,X
2795  D3CE  AD 46 FB           LDA DIRTRK
2796  D3D1  85 13              STA TRACK
```

```
LINE# LOC   CODE        LINE


2797  D3D3  A9 00                LDA #0
2798  D3D5  85 14                STA SECTOR
2799  D3D7  20 5F F9             JSR SETBJ
2800  D3DA  20 5C D7             JSR DRTRD
2801  D3DD  A6 12                LDX DRVNUM
2802  D3DF  A9 00                LDA #0
2803  D3E1  9D A6 10             STA WPSW,X     ;CLEAR WP SWITCH
2804  D3E4  9D A1 43             STA NODRV,X
2805  D3E7                ;
2806  D3E7                ; CALCULATE FREE BLOCKS
2807  D3E7                ;
2808  D3E7                NF05
2809  D3E7  A6 12                LDX DRVNUM
2810  D3E9  A9 00                LDA #00
2811  D3EB  9D 9D 43             STA NDBL,X
2812  D3EE  9D 9F 43             STA NDBH,X
2813  D3F1  85 13                STA TRACK
2814  D3F3                NF10
2815  D3F3  E6 13                INC TRACK
2816  D3F5  A5 13                LDA TRACK
2817  D3F7  CD B0 10             CMP MAXTRK
2818  D3FA  90 01                BCC NF20
2819  D3FC  60                   RTS
2820  D3FD                NF20
2821  D3FD  CD 46 FB             CMP DIRTRK
2822  D400  F0 F1                BEQ NF10
2823  D402  20 0A F9             JSR SETMAP
2824  D405  A6 12                LDX DRVNUM
2825  D407  B1 02                LDA (BMPNT),Y
2826  D409  18                   CLC
2827  D40A  7D 9D 43             ADC NDBL,X
2828  D40D  9D 9D 43             STA NDBL,X
2829  D410  90 E1                BCC NF10
2830  D412  FE 9F 43             INC NDBH,X
2831  D415  4C F3 D3             JMP NF10
2832  D418                ;
2833  D418                ;
2834  D418                ; START DOUBLE BUFFERING
2835  D418                ; USE TRACK,SECTOR AS STARTING BLOACK
2836  D418                ;
2837  D418                STRDBL
2838  D418  20 D9 D9             JSR SETHDR
2839  D41B  20 3C D4             JSR RDBUF
2840  D41E  20 AF D8             JSR WATJOB
2841  D421  20 AE D4             JSR GETBYT
2842  D424  85 13                STA TRACK
2843  D426  20 AE D4             JSR GETBYT
2844  D429  85 14                STA SECTOR
2845  D42B  A5 13                LDA TRACK
2846  D42D  D0 01                BNE STR1
2847  D42F  60                   RTS
2848  D430  20 12 D3       STR1  JSR DBLBUF
2849  D433  20 D9 D9             JSR SETHDR
2850  D436  20 3C D4             JSR RDBUF
2851  D439  4C 12 D3             JMP DBLBUF
```

```
LINE# LOC   CODE          LINE

2852  D43C                ;
2853  D43C                ; START A READ JOB ON TRACK,SECTOR
2854  D43C                ;
2855  D43C  A9 80         RDBUF  LDA #READ
2856  D43E  D0 02                BNE STRTIT
2857  D440                ;
2858  D440                ; START A WRITE JOB ON TRACK,SECTOR
2859  D440                ;
2860  D440  A9 90         WRTBUF LDA #WRITE
2861  D442  8D 3E 43      STRTIT STA CMD
2862  D445  20 D5 E7             JSR GETACT
2863  D448  AA                   TAX
2864  D449  20 13 D8             JSR SETLJB
2865  D44C  8A                   TXA
2866  D44D  48                   PHA
2867  D44E  0A                   ASL A
2868  D44F  AA                   TAX
2869  D450  A9 00                LDA #0
2870  D452  95 29                STA BUFTAB,X
2871  D454  20 9C D4             JSR TYPFIL
2872  D457  C9 04                CMP #4
2873  D459  B0 06                BCS WRTC1       ;NOT SEQUENTIAL TYPE
2874  D45B                ;
2875  D45B  F6 59                INC NBKL,X
2876  D45D  D0 02                BNE WRTC1
2877  D45F  F6 61                INC NBKH,X
2878  D461  68            WRTC1  PLA
2879  D462  AA                   TAX
2880  D463  60                   RTS
2881  D464                ;
2882  D464                ;
2883  D464                ;****************************
2884  D464                ;*
2885  D464                ;*   FNDRCH
2886  D464                ;******************************
2887  D464                ;*
2888  D464                ;*
2889  D464  A5 16         FNDRCH LDA SA
2890  D466  C9 13                CMP #MAXSA+1
2891  D468  90 02                BCC FNDC20
2892  D46A  29 0F                AND #$F
2893  D46C                FNDC20
2894  D46C  C9 0F                CMP #CMDSA
2895  D46E  D0 02                BNE FNDC25
2896  D470  A9 10                LDA #ERRSA
2897  D472                FNDC25
2898  D472  AA                   TAX
2899  D473  38                   SEC
2900  D474  B5 AC                LDA LINTAB,X
2901  D476  30 06                BMI FNDC30
2902  D478  29 0F                AND #$F
2903  D47A  85 15                STA LINDX
2904  D47C  AA                   TAX
2905  D47D  18                   CLC
2906  D47E  60            FNDC30 RTS
```

```
LINE# LOC   CODE        LINE


2962  D4BE  D0 04                   BNE GETB2
2963  D4C0  A9 FF                   LDA #$FF
2964  D4C2  95 29                   STA BUFTAB,X
2965  D4C4  68              GETB2   PLA
2966  D4C5  F6 29                   INC BUFTAB,X
2967  D4C7  60                      RTS
2968  D4C8  A1 29           GETB1   LDA (BUFTAB,X)
2969  D4CA  F6 29                   INC BUFTAB,X
2970  D4CC  60                      RTS
2971  D4CD                  ;
2972  D4CD                  ;
2973  D4CD                  ; READ A CHAR FROM FILE AND READ NEXT
2974  D4CD                  ; BLOCK OF FILE IF NEEDED.
2975  D4CD                  ; SET CHNRDY=EOI IF END OF FILE
2976  D4CD                  ;
2977  D4CD  20 AE D4        RDBYT   JSR GETBYT
2978  D4D0  D0 36                   BNE RD3
2979  D4D2  85 18                   STA DATA

2981  D4D4  B9 C7 00        RD0     LDA LSTCHR,Y
2982  D4D7  F0 08                   BEQ RD1
2983  D4D9  A9 80                   LDA #EOIOUT
2984  D4DB  99 A2 00        RD01    STA CHNRDY,Y
2985  D4DE  A5 18                   LDA DATA
2986  D4E0  60                      RTS
2987  D4E1  20 12 D3        RD1     JSR DBLBUF
2988  D4E4  A9 00                   LDA #0
2989  D4E6  20 C6 D7                JSR SETPNT
2990  D4E9  20 AE D4                JSR GETBYT
2991  D4EC  C9 00                   CMP #0
2992  D4EE  F0 19                   BEQ RD4
2993  D4F0  85 13                   STA TRACK
2994  D4F2  20 AE D4                JSR GETBYT
2995  D4F5  85 14                   STA SECTOR
2996  D4F7  20 12 D3                JSR DBLBUF
2997  D4FA  20 4A D5                JSR SETDRN
2998  D4FD  20 D9 D9                JSR SETHDR
2999  D500  20 3C D4                JSR RDBUF
3000  D503  20 12 D3                JSR DBLBUF
3001  D506  A5 18                   LDA DATA
3002  D508  60              RD3     RTS
3003  D509  20 AE D4        RD4     JSR GETBYT
3004  D50C  A4 15                   LDY LINDX
3005  D50E  99 C7 00                STA LSTCHR,Y
3006  D511  A5 18                   LDA DATA
3007  D513  60                      RTS



3010  D514                  ; WRITE A CHAR TO CHANL AND WRITE
3011  D514                  ; BUFFER OUT TO DISK IF ITS FULL
3012  D514                  ;
3013  D514  20 73 D3        WRTBYT JSR PUTBYT
3014  D517  F0 01                   BEQ WRT0
3015  D519  60                      RTS
```

```
LINE# LOC    CODE        LINE


3017  D51A   20 4A D5    WRTO   JSR SETDRN
3018  D51D   20 FB F9           JSR NXTTS
3019  D520   A9 00              LDA #0
3020  D522   20 C6 D7           JSR SETPNT
3021  D525   A5 13              LDA TRACK
3022  D527   20 73 D3           JSR PUTBYT
3023  D52A   A5 14              LDA SECTOR
3024  D52C   20 73 D3           JSR PUTBYT
3025  D52F   20 40 D4           JSR WRTBUF
3026  D532   20 12 D3           JSR DBLBUF
3027  D535   20 D9 D9           JSR SETHDR
3028  D538   A9 02              LDA #2
3029  D53A   4C C6 D7           JMP SETPNT
3030  D53D               ;
3031  D53D               ; INC POINTER OF ACTIVE BUFFER
3032  D53D               ; BY .A
3033  D53D               ;
3034  D53D               ;
3035  D53D               INCPTR                   ; SCOTT PATCH
3036  D53D   85 04       INCPNT STA TEMP
3037  D53F   20 E6 D7           JSR GETPNT
3038  D542   18                 CLC
3039  D543   65 04              ADC TEMP
3040  D545   95 29              STA BUFTAB,X
3041  D547   85 27              STA DIRBUF
3042  D549   60                 RTS
3043  D54A               ; SET DRVNUM TO DRIVE INDICATED BY
3044  D54A               ; LSTJOB OF ACTIVE BUFFER
3045  D54A               ;
3046  D54A   20 D5 E7    SETDRN JSR GETACT
3047  D54D   AA                 TAX
3048  D54E   BD 50 43           LDA LSTJOB,X
3049  D551   29 01              AND #1
3050  D553   85 12              STA DRVNUM
3051  D555   60                 RTS
3052  D556                      .END
3053  D556                      .LIB TST3
```

```
LINE# LOC   CODE        LINE


3055  D556              ;
3056  D556              ;
3057  D556              ;  .A=#BUFFERS NEEDED
3058  D556              ;  SETS UP BUFFER # AND ALLOCATES LINDX
3059  D556              GETWCH
3060  D556  38                 SEC            ;SET .C=1 INDICATE WRITE
3061  D557  B0 01              BCS GETR2
3062  D559              GETRCH
3063  D559  18                 CLC            ;SET .C=0 INDICATE READ
3064  D55A              GETR2
3065  D55A  08                 PHP            ;SAVE R/W FLAG (.C)
3066  D55B  85 04              STA TEMP       ;SAVE #BUFS NEEDED
3067  D55D  20 9D D5           JSR FRECHN     ;FREE ANY CHANNELS
3068  D560  20 7E D6           JSR FNDLNX     ;  GET NEXT LINDX OPEN
3069  D563  85 15              STA LINDX
3070  D565  A6 16              LDX SA
3071  D567  28                 PLP
3072  D568  90 02              BCC GETR55
3073  D56A  09 80       GETR52 ORA #$80
3074  D56C  95 AC       GETR55 STA LINTAB,X     ;SAVE LINDX IN LINTAB
3075  D56E              GETBF
3076  D56E  29 3F              AND #$3F
3077  D570  A8                 TAY            ;NOW GET THE BUFFERS
3078  D571  A9 FF              LDA #$FF
3079  D573  99 79 00           STA SS,Y
3080  D576  99 49 00           STA BUF0,Y
3081  D579  99 51 00           STA BUF1,Y
3082  D57C  C6 04       GETR3  DEC TEMP
3083  D57E  30 1C              BMI GETR4
3084  D580  20 02 D6           JSR GETBUF
3085  D583  10 08              BPL GETR5
3086  D585  20 CE D5    GBERR  JSR RELBUF       ;  ERROR ,REL BUFS
3087  D588  A9 70              LDA #NOCHNL
3088  D58A  4C 96 C4           JMP CMDERR
3089  D58D  99 49 00    GETR5  STA BUF0,Y
3090  D590  C6 04              DEC TEMP
3091  D592  30 08              BMI GETR4
3092  D594  20 02 D6           JSR GETBUF
3093  D597  30 EC              BMI GBERR
3094  D599  99 51 00           STA BUF1,Y
3095  D59C  60          GETR4  RTS
3096  D59D              ;
3097  D59D              ;  FREE CHANL ASSOCIATED WITH SA
3098  D59D              ;  FREE READ AND WRITE CHANLS
3099  D59D              ;  DONT FREE CHANL 15
3100  D59D  A5 16       FRECHN LDA SA
3101  D59F  C9 0F              CMP #$F
3102  D5A1  D0 01              BNE FRECO
3103  D5A3  60                 RTS

3105  D5A4              FRECO
3106  D5A4              FRERD
3107  D5A4  A6 16       FREWRT LDX SA
3108  D5A6  B5 AC              LDA LINTAB,X
3109  D5A8  C9 FF              CMP #$FF
```

```
LINE# LOC   CODE        LINE


3110  D5AA  F0 21              BEQ FRE25
3111  D5AC  29 3F              AND #$3F
3112  D5AE  85 15              STA LINDX
3113  D5B0  A9 FF              LDA #$FF
3114  D5B2  95 AC              STA LINTAB,X
3115  D5B4  A6 15              LDX LINDX
3116  D5B6  A9 00              LDA #00
3117  D5B8  95 A2              STA CHNRDY,X
3118  D5BA  20 CE D5           JSR RELBUF
3119  D5BD              ;
3120  D5BD              ; RELEASE THE LINDX
3121  D5BD              ;
3122  D5BD  A6 15       RELINX LDX LINDX
3123  D5BF  A9 01              LDA #1
3124  D5C1  CA          REL15  DEX
3125  D5C2  30 03              BMI REL10
3126  D5C4  0A                 ASL A
3127  D5C5  D0 FA              BNE REL15
3128  D5C7  0D 4B 43    REL10  ORA LINUSE      ;1=FREE 0=USED
3129  D5CA  8D 4B 43           STA LINUSE
3130  D5CD  60          FRE25  RTS
3131  D5CE              ; GIVEN SA , FREE ITS READ CHANL
3132  D5CE              ;
3133  D5CE              ;
3134  D5CE              ; RELEASE BUFFERS (LINDX)
3135  D5CE              ;
3136  D5CE  A6 15       RELBUF LDX LINDX
3137  D5D0  B5 49              LDA BUF0,X
3138  D5D2  C9 FF              CMP #$FF
3139  D5D4  F0 09              BEQ REL1
3140  D5D6  48                 PHA
3141  D5D7  A9 FF              LDA #$FF
3142  D5D9  95 49              STA BUF0,X
3143  D5DB  68                 PLA
3144  D5DC  20 39 D6           JSR FREBUF
3145  D5DF  A6 15       REL1   LDX LINDX
3146  D5E1  B5 51              LDA BUF1,X
3147  D5E3  C9 FF              CMP #$FF
3148  D5E5  F0 09              BEQ REL2
3149  D5E7  48                 PHA
3150  D5E8  A9 FF              LDA #$FF
3151  D5EA  95 51              STA BUF1,X
3152  D5EC  68                 PLA
3153  D5ED  20 39 D6           JSR FREBUF
3154  D5F0  A6 15       REL2   LDX LINDX
3155  D5F2  B5 79              LDA SS,X
3156  D5F4  C9 FF              CMP #$FF
3157  D5F6  F0 09              BEQ REL3
3158  D5F8  48                 PHA
3159  D5F9  A9 FF              LDA #$FF
3160  D5FB  95 79              STA SS,X
3161  D5FD  68                 PLA
3162  D5FE  20 39 D6           JSR FREBUF
3163  D601  60          REL3   RTS
3164  D602              ;
```

```
LINE# LOC   CODE         LINE


3165  D602               ; GET A FREE BUFFER #
3166  D602               ;
3167  D602  A9 FF        GETBUF LDA #$FF
3168  D604  85 05               STA TEMP+1
3169  D606  A2 0F               LDX #15
3170  D608  2E 40 43     GETBU1 ROL BUFUSE
3171  D60B  2E 41 43            ROL BUFUSE+1
3172  D60E  B0 05               BCS GETBU2
3173  D610  86 05               STX TEMP+1
3174  D612  38                  SEC
3175  D613  B0 19               BCS GETBU3
3176  D615  CA           GETBU2 DEX
3177  D616  10 F0               BPL GETBU1
3178  D618  2E 40 43            ROL BUFUSE
3179  D61B  2E 41 43            ROL BUFUSE+1
3180  D61E  A6 05        GETBU4 LDX TEMP+1
3181  D620  30 0A               BMI GETB5
3182  D622  A9 00               LDA #0
3183  D624  9D 03 10            STA JOBS,X
3184  D627  A5 12               LDA DRVNUM
3185  D629  9D 50 43            STA LSTJOB,X
3186  D62C  8A           GETB5  TXA
3187  D62D  60                  RTS
3188  D62E  2E 40 43     GETBU3 ROL BUFUSE
3189  D631  2E 41 43            ROL BUFUSE+1
3190  D634  CA                  DEX
3191  D635  10 F7               BPL GETBU3
3192  D637  30 E5               BMI GETBU4
3193  D639               ; ALLOCATE
3194  D639               ;
3195  D639               ;
3196  D639  29 0F        FREBUF AND #$F
3197  D63B  A8                  TAY
3198  D63C  C8                  INY
3199  D63D  A2 10               LDX #16
3200  D63F  6E 41 43     FREB1  ROR BUFUSE+1
3201  D642  6E 40 43            ROR BUFUSE
3202  D645  88                  DEY
3203  D646  D0 01               BNE FREB2
3204  D648  18                  CLC
3205  D649  CA           FREB2  DEX
3206  D64A  10 F3               BPL FREB1
3207  D64C  60                  RTS
3208  D64D               ;
3209  D64D               ;
3210  D64D  A9 0E        CLRCHN LDA #14
3211  D64F  85 16               STA SA
3212  D651  20 9D D5     CLRC1  JSR FRECHN
3213  D654  C6 16               DEC SA
3214  D656  D0 F9               BNE CLRC1
3215  D658  60                  RTS
3216  D659               ;
3217  D659  A9 0E        CLDCHN LDA #14
3218  D65B  85 16               STA SA
3219  D65D  A6 16        CLSD   LDX SA
```

```
LINE# LOC   CODE        LINE

3220  D65F  B5 AC              LDA LINTAB,X
3221  D661  C9 FF              CMP #$FF
3222  D663  F0 14              BEQ CLD2
3223  D665  29 3F              AND #$3F
3224  D667  85 15              STA LINDX
3225  D669  20 D5 E7           JSR GETACT
3226  D66C  AA                 TAX
3227  D66D  BD 50 43           LDA LSTJOB,X
3228  D670  29 01              AND #1
3229  D672  C5 12              CMP DRVNUM
3230  D674  D0 03              BNE CLD2
3231  D676  20 9D D5           JSR FRECHN
3232  D679  C6 16       CLD2   DEC SA
3233  D67B  10 E0              BPL CLSD
3234  D67D  60                 RTS
3235  D67E              ;
3236  D67E              ; FIND A FREE LINDX TO USE
3237  D67E              ; MARK AS USED IN LINUSE
3238  D67E              ;
3239  D67E  A0 00       FNDLNX LDY #0
3240  D680  A9 01              LDA #1
3241  D682  2C 4B 43    FND10  BIT LINUSE      ;1=FREE 0=USED
3242  D685  D0 09              BNE FND30
3243  D687  C8                 INY
3244  D688  0A                 ASL A
3245  D689  D0 F7              BNE FND10
3246  D68B  A9 70              LDA #NOCHNL    ;NO FREE LINDX AVAILABLE
3247  D68D  4C 96 C4           JMP CMDERR
3248  D690  49 FF       FND30  EOR #$FF        ;TOGGLE BIT MASK
3249  D692  2D 4B 43           AND LINUSE     ;MARK BIT USED
3250  D695  8D 4B 43           STA LINUSE
3251  D698  98                 TYA            ;RETURN LINDX IN .A
3252  D699  60                 RTS
3253  D69A                     .END
3254  D69A                     .LIB TST4
```

```
LINE# LOC    CODE          LINE

3256  D69A                 ;
3257  D69A                 ;  GET NEXT CHAR FROM A CHANL
3258  D69A                 ;
3259  D69A                 GBYTE                      ;GET THROUGH INTERNAL CHANN
3260  D69A   20 64 D4              JSR FNDRCH
3261  D69D   20 DC F9              JSR SETLDS
3262  D6A0   20 A8 D6              JSR GET
3263  D6A3   A6 15                 LDX LINDX
3264  D6A5   B5 BF                 LDA CHNDAT,X
3265  D6A7   60                    RTS
3266  D6A8                 ;
3267  D6A8                 ;
3268  D6A8   A6 15         GET     LDX LINDX
3269  D6AA   20 9C D4              JSR TYPFIL        ;GET FILE TYPE
3270  D6AD   D0 03                 BNE GET00
3271  D6AF                 ;
3272  D6AF   4C 49 E9              JMP RDREL
3273  D6B2                 GET00
3274  D6B2   A5 16                 LDA SA
3275  D6B4   C9 0F                 CMP #$F
3276  D6B6   F0 59                 BEQ GETERC
3277  D6B8                 ;
3278  D6B8   B5 A2                 LDA CHNRDY,X      ;WAS LAST CHAR JUST SENT
3279  D6BA   29 08                 AND #EOISND       ; JUST SENT EOI
3280  D6BC   D0 13                 BNE GET1          ;NOPE NOT THIS TIME
3281  D6BE                 ;
3282  D6BE   20 9C D4              JSR TYPFIL
3283  D6C1   C9 07                 CMP #DIRTYP
3284  D6C3   D0 07                 BNE GET0          ;NOT DIRECT TYPE
3285  D6C5                 ;
3286  D6C5   A9 89                 LDA #RNDRDY       ; DIRECT FILE REMAINS ACTIVE
3287  D6C7   95 A2                 STA CHNRDY,X      ;TALKER LISTENER NOEOI
3288  D6C9   4C DC D6              JMP RNDGET        ; PREPARE THE NEXT CHAR
3289  D6CC                 ;
3290  D6CC   A9 00         GET0    LDA #NOTRDY        ;LST CHAR SENT,NOT READY
3291  D6CE   95 A2                 STA CHNRDY,X
3292  D6D0   60                    RTS
3293  D6D1                 ;
3294  D6D1                 ;
3295  D6D1   A5 16         GET1    LDA SA             ;TEST IF A LOAD
3296  D6D3   F0 31                 BEQ GET6          ;ITS A LOAD SA OF 0
3297  D6D5                 ;
3298  D6D5   20 9C D4      GET2    JSR TYPFIL         ; TST FOR RND FILE
3299  D6D8   C9 04                 CMP #4
3300  D6DA   90 22                 BCC SEQGET        ;SEQ FILE CHAR GET
3301  D6DC                 ;
3302  D6DC   20 A6 D4      RNDGET  JSR GETPRE         ;DIRECT FILE GET
3303  D6DF   B5 29                 LDA BUFTAB,X
3304  D6E1   D9 C7 00              CMP LSTCHR,Y      ;UP TO LST CHAR YET
3305  D6E4   D0 04                 BNE RNGET1        ;NOT YET
3306  D6E6                 ;
3307  D6E6   A9 00                 LDA #0            ;READ THE WHOLE THING
3308  D6E8   95 29                 STA BUFTAB,X      ;WRAP PNTR TO 0
3309  D6EA                 RNGET1
3310  D6EA   F6 29                 INC BUFTAB,X      ;GET THE NEXT CHAR
```

```
LINE# LOC   CODE         LINE

3311  D6EC              RNGET2
3312  D6EC  A1 29              LDA (BUFTAB,X)
3313  D6EE              RNGET4
3314  D6EE  99 BF 00           STA CHNDAT,Y    ;SAVE CHAR IN
3315  D6F1  B5 29              LDA BUFTAB,X
3316  D6F3  D9 C7 00           CMP LSTCHR,Y
3317  D6F6  D0 05              BNE RNGET3
3318  D6F8              ;
3319  D6F8  A9 81              LDA #RNDEOI
3320  D6FA  99 A2 00           STA CHNRDY,Y    ;THIS IS LAST CHAR
3321  D6FD  60          RNGET3 RTS              ;SEND EOI WITH IT
3322  D6FE              ;
3323  D6FE              ;
3324  D6FE  20 CD D4    SEQGET JSR RDBYT        ;READ THE NEXT BYTE
3325  D701  A6 15       GET3   LDX LINDX
3326  D703  95 BF              STA CHNDAT,X    ;STORE IN CHNDAT
3327  D705  60                 RTS
3328  D706              ;
3329  D706              ;
3330  D706  AD 49 43    GET6   LDA DIRLST       ;TEST IF DIR LSTING
3331  D709  F0 F3              BEQ SEQGET      ;NO, ITS NOT
3332  D70B  20 57 F5           JSR GETDIR      ;YES IT IS,GET DIR CHAR
3333  D70E  4C 01 D7           JMP GET3
3334  D711              ;
3335  D711              GETERC
3336  D711  20 E6 D7           JSR GETPNT
3337  D714  C9 DB              CMP #<ERRBUF-1
3338  D716  D0 18              BNE GE10
3339  D718  A5 28              LDA DIRBUF+1
3340  D71A  C9 43              CMP #>ERRBUF
3341  D71C  D0 12              BNE GE10
3342  D71E              ;
3343  D71E  A9 0D              LDA #CR
3344  D720  85 18              STA DATA
3345  D722  20 F2 F9           JSR ERROFF
3346  D725  A9 00              LDA #0
3347  D727  20 76 EF           JSR ERRTS0
3348  D72A  C6 47              DEC CB+2
3349  D72C  A9 80              LDA #EOIOUT
3350  D72E  D0 12              BNE GE30         ;(JUMP)
3351  D730              ;
3352  D730              GE10
3353  D730  20 AE D4           JSR GETBYT
3354  D733  85 18              STA DATA
3355  D735  D0 09              BNE GE20
3356  D737              GE15
3357  D737  A9 DB              LDA #<ERRBUF-1
3358  D739  20 C6 D7           JSR SETPNT
3359  D73C  A9 43              LDA #>ERRBUF
3360  D73E  95 2A              STA BUFTAB+1,X
3361  D740              GE20
3362  D740  A9 88              LDA #RDYTLK
3363  D742              GE30
3364  D742  85 A9              STA CHNRDY+ERRCHN
3365  D744  A5 18              LDA DATA
```

```
LINE# LOC   CODE        LINE

3366  D746  85 C6              STA CHNDAT+ERRCHN
3367  D748  60                 RTS
3368  D749              ;
3369  D749              ;
3370  D749              ; READ NEXT BUFFER OF A FILE
3371  D749              ; FOLLOW LINKS IN FIRST TWO BYTES
3372  D749              ; END OF FILE IF 1ST BYTE=0
3373  D749              ; 2ND CHAR LENGTH
3374  D749              ;
3375  D749  20 D5 E7    NXTBUF JSR GETACT
3376  D74C  0A                 ASL A
3377  D74D  AA                 TAX
3378  D74E  A9 00              LDA #0
3379  D750  95 29              STA BUFTAB,X
3380  D752  A1 29              LDA (BUFTAB,X)
3381  D754  F0 05              BEQ NXTB1
3382  D756  D6 29              DEC BUFTAB,X
3383  D758  4C CD D4           JMP RDBYT
3384  D75B  60          NXTB1  RTS
3385  D75C              ;
3386  D75C              ; DIRECT BLOCK READ
3387  D75C              ;
3388  D75C              ;
3389  D75C  A9 80       DRTRD  LDA #READ
3390  D75E  D0 02              BNE DRT
3391  D760              ; DIRECT BLOCK WRITE
3392  D760              ;
3393  D760              ; DIRECT BLOCK WRITE
3394  D760  A9 90       DRTWRT LDA #WRITE
3395  D762  05 12       DRT    ORA DRVNUM
3396  D764  8D 3E 43           STA CMD
3397  D767  A5 AB              LDA JOBNUM
3398  D769  20 DC D9           JSR SETH
3399  D76C  A6 AB              LDX JOBNUM
3400  D76E  4C A9 D8           JMP DOIT2
3401  D771              ; OPEN INTERNAL READ CHANL (SA=16)
3402  D771              ;
3403  D771              OPNIRD
3404  D771  A9 01              LDA #1
3405  D773              OPNTYP
3406  D773  85 CF              STA TYPE
3407  D775  A9 11              LDA #IRSA
3408  D777  85 16              STA SA
3409  D779  20 7D E4           JSR OPNRCH
3410  D77C  A9 02              LDA #2
3411  D77E  4C C6 D7           JMP SETPNT
3412  D781              ; OPEN INTERNAL WRITE CHANL (SA=16)
3413  D781              ;
3414  D781              ;
3415  D781  A9 12       OPNIWR LDA #IWSA
3416  D783  85 16              STA SA
3417  D785  4C 1C E5           JMP OPNWCH
3418  D788              ;
3419  D788              ; ALLOCATE NEXT DIRECTORY BLOCK ON 18
3420  D788              ;   AND MARK AS USED IN BAM
```

```
LINE# LOC   CODE        LINE


3421  D788              ;
3422  D788  20 7B E6    NXDRBK JSR CURBLK
3423  D78B  A9 01              LDA #1
3424  D78D  85 04              STA TEMP
3425  D78F  AD F4 10           LDA SECINC
3426  D792  48                 PHA
3427  D793  A9 03              LDA #3          ;   INCR SECTOR BY 3 IN DIREC
3428  D795  8D F4 10           STA SECINC
3429  D798  20 02 FA           JSR NXTDS
3430  D79B  68                 PLA
3431  D79C  8D F4 10           STA SECINC
3432  D79F  A9 00              LDA #0
3433  D7A1  20 C6 D7           JSR SETPNT
3434  D7A4  A5 13              LDA TRACK
3435  D7A6  20 73 D3           JSR PUTBYT
3436  D7A9  A5 14              LDA SECTOR
3437  D7AB  20 73 D3           JSR PUTBYT
3438  D7AE  20 40 D4           JSR WRTBUF
3439  D7B1  20 AF D8           JSR WATJOB
3440  D7B4  A9 00              LDA #0
3441  D7B6  20 C6 D7           JSR SETPNT
3442  D7B9  20 73 D3    NXDB1  JSR PUTBYT
3443  D7BC  D0 FB              BNE NXDB1
3444  D7BE  20 73 D3           JSR PUTBYT
3445  D7C1  A9 FF              LDA #$FF
3446  D7C3  4C 73 D3           JMP PUTBYT
3447  D7C6              ;
3448  D7C6              ;
3449  D7C6              ;  . A=NEW PNTR VALUE
3450  D7C6              ;
3451  D7C6  85 04       SETPNT STA TEMP
3452  D7C8  20 D5 E7           JSR GETACT
3453  D7CB  0A                 ASL A
3454  D7CC  AA                 TAX
3455  D7CD  B5 2A              LDA BUFTAB+1,X
3456  D7CF  85 28              STA DIRBUF+1
3457  D7D1  A5 04              LDA TEMP
3458  D7D3  95 29              STA BUFTAB,X
3459  D7D5  85 27              STA DIRBUF
3460  D7D7  60                 RTS
3461  D7D8              ;
3462  D7D8              ;
3463  D7D8              ;
3464  D7D8              ; FREE THE INTERNAL CHANL (SA=16)
3465  D7D8              ;
3466  D7D8  A9 11       FREICH LDA #IRSA
3467  D7DA  85 16              STA SA
3468  D7DC  20 9D D5           JSR FRECHN
3469  D7DF  A9 12              LDA #IWSA
3470  D7E1  85 16              STA SA
3471  D7E3  4C 9D D5           JMP FRECHN
3472  D7E6              ;
3473  D7E6              ; READ THE ACTIVE BUFFER POINTER
3474  D7E6              ;
3475  D7E6  20 D5 E7    GETPNT JSR GETACT
```

```
LINE# LOC   CODE        LINE


3476  D7E9              SETDIR
3477  D7E9  0A          GP1     ASL A
3478  D7EA  AA                  TAX
3479  D7EB  B5 2A               LDA BUFTAB+1,X
3480  D7ED  85 28               STA DIRBUF+1
3481  D7EF  B5 29               LDA BUFTAB,X
3482  D7F1  85 27               STA DIRBUF
3483  D7F3  60                  RTS
3484  D7F4              ;
3485  D7F4              ; DIRECT READ BYTE, .A=BYTE# TO READ
3486  D7F4              ;
3487  D7F4  85 06       DRDBYT STA TEMP+2
3488  D7F6  20 D5 E7            JSR GETACT
3489  D7F9  AA                  TAX
3490  D7FA  BD 04 D8            LDA BUFIND,X
3491  D7FD  85 07               STA TEMP+3
3492  D7FF  A0 00               LDY #0
3493  D801  B1 06               LDA (TEMP+2),Y
3494  D803  60                  RTS
3495  D804              ; INDEX TABLE CONTAINING HIGH
3496  D804              ; BYTE ADDRESS OF BUFFERS
3497  D804              ;
3498  D804              BUFIND
3499  D804  11                  .BYTE $11,$12,$13
3499  D805  12
3499  D806  13
3500  D807  20                  .BYTE $20,$21,$22,$23
3500  D808  21
3500  D809  22
3500  D80A  23
3501  D80B  30                  .BYTE $30,$31,$32,$33
3501  D80C  31
3501  D80D  32
3501  D80E  33
3502  D80F  40                  .BYTE $40,$41,$42,$43
3502  D810  41
3502  D811  42
3502  D812  43
3503  D813                      .END
3504  D813                      .LIB JOBS
```

```
LINE# LOC   CODE        LINE

3506  D813              ;
3507  D813              ; USE LASTJOB FOR DRIVE #
3508  D813              ; CMD IS USED FOR JOB COMMAND
3509  D813              ;
3510  D813              SETLJB
3511  D813  BD 50 43            LDA LSTJOB,X
3512  D816  29 01               AND #1
3513  D818  0D 3E 43            ORA CMD
3514  D81B              ;
3515  D81B              ; SET JOB UP AND CHECK T&S
3516  D81B              ;   .A=COMMAND FOR JOBS
3517  D81B              ;   .X=JOB NUMBER
3518  D81B              ;
3519  D81B              SETJOB
3520  D81B  48                  PHA
3521  D81C  86 AB               STX JOBNUM
3522  D81E  8A                  TXA
3523  D81F  0A                  ASL A
3524  D820  0A                  ASL A
3525  D821  0A                  ASL A
3526  D822  AA                  TAX
3527  D823  BD 24 10            LDA HDRS+3,X
3528  D826  8D 3E 43            STA CMD         ;SAVE SECTOR
3529  D829  BD 23 10            LDA HDRS+2,X
3530  D82C  F0 2D               BEQ TSERR
3531  D82E              ;
3532  D82E  CD B0 10            CMP MAXTRK
3533  D831  B0 28               BCS TSERR       ;TRACK TOO LARGE
3534  D833              ;
3535  D833  AA                  TAX
3536  D834  68                  PLA             ;CHECK FOR WRITE
3537  D835  48                  PHA
3538  D836  29 F0               AND #$F0
3539  D838  C9 90               CMP #WRITE
3540  D83A  D0 53               BNE SJB1        ;NOT WRITE,SKIP CHECK
3541  D83C  68                  PLA
3542  D83D  48                  PHA
3543  D83E  4A                  LSR A
3544  D83F  B0 05               BCS SJB2        ;DRIVE 1
3545  D841              ;
3546  D841  AD 02 41            LDA BAM0+2      ;GET VERSION #
3547  D844  90 03               BCC SJB3
3548  D846              SJB2
3549  D846  AD 02 42            LDA BAM1+2      ;GET DRIVE 1 VER#
3550  D849              SJB3
3551  D849  F0 05               BEQ SJB4        ;NO # IS OK, TOO
3552  D84B  CD 9F 10            CMP VERNUM
3553  D84E  D0 37               BNE VNERR       ;NOT SAME VERNUM #
3554  D850              ;
3555  D850              SJB4
3556  D850  8A                  TXA             ;RESTORE TRACK #
3557  D851  20 2C FB            JSR MAXSEC
3558  D854  CD 3E 43            CMP CMD
3559  D857  F0 02               BEQ TSERR
3560  D859  B0 34               BCS SJB1        ;SECTOR IS OK!
```

```
LINE# LOC   CODE        LINE

3561  D85B              ;
3562  D85B              ;
3563  D85B              ; ILLEGAL TRACK AND SECTOR
3564  D85B              ;
3565  D85B              TSERR
3566  D85B  20 63 D8            JSR HED2TS
3567  D85E              TSER1
3568  D85E  A9 66               LDA #BADTS
3569  D860  4C EA EE            JMP CMDER2
3570  D863              ;
3571  D863              ;
3572  D863              HED2TS
3573  D863  A5 AB               LDA JOBNUM
3574  D865  0A                  ASL A
3575  D866  0A                  ASL A
3576  D867  0A                  ASL A
3577  D868  AA                  TAX
3578  D869  BD 23 10            LDA HDRS+2,X
3579  D86C  85 13               STA TRACK
3580  D86E  BD 24 10            LDA HDRS+3,X
3581  D871  85 14               STA SECTOR
3582  D873  60                  RTS
3583  D874              ;
3584  D874              ;
3585  D874              TSCHK
3586  D874  A5 13               LDA TRACK
3587  D876  F0 E6               BEQ TSER1
3588  D878  CD B0 10            CMP MAXTRK
3589  D87B  B0 E1               BCS TSER1
3590  D87D              ;
3591  D87D  20 2C FB            JSR MAXSEC
3592  D880  C5 14               CMP SECTOR
3593  D882  F0 DA               BEQ TSER1
3594  D884  90 D8               BCC TSER1
3595  D886  60                  RTS
3596  D887              ;
3597  D887              VNERR
3598  D887  20 63 D8            JSR HED2TS
3599  D88A  A9 73               LDA #CBMV2      ;WRITE TO WRONG VERSION
3600  D88C  4C EA EE            JMP CMDER2
3601  D88F              ;
3602  D88F              SJB1
3603  D88F  A6 AB               LDX JOBNUM
3604  D891  68                  PLA
3605  D892  8D 3E 43            STA CMD
3606  D895  9D 03 10            STA JOBS,X
3607  D898  9D 50 43            STA LSTJOB,X
3608  D89B  60                  RTS
3609  D89C              ;
3610  D89C              ;
3611  D89C              ; DO JOB IN .A, SET UP ERROR COUNT
3612  D89C              ; AND LSTJOB. RETURN WHEN JOB DONE OK
3613  D89C              ; JMP TO ERROR IF ERROR RETURNS
3614  D89C              ;
3615  D89C              DOREAD
```

```
LINE# LOC   CODE         LINE

3616  D89C  A9 80                 LDA #READ
3617  D89E  D0 02                 BNE DOJOB        ;BRA
3618  D8A0               DOWRIT
3619  D8A0  A9 90                 LDA #WRITE
3620  D8A2               DOJOB
3621  D8A2  05 12                 ORA DRVNUM
3622  D8A4  A6 AB                 LDX JOBNUM
3623  D8A6               ;
3624  D8A6  8D 3E 43     DOIT     STA CMD
3625  D8A9  AD 3E 43     DOIT2    LDA CMD
3626  D8AC  20 1B D8              JSR SETJOB
3627  D8AF               ;        JMP WATJOB
3628  D8AF               ;
3629  D8AF               ;  WAIT UNTIL JOB(.X) IS DONE
3630  D8AF               ;  RETURN WHEN DONE
3631  D8AF               ;
3632  D8AF  20 BC D8     WATJOB   JSR TSTJOB
3633  D8B2  B0 FB                 BCS WATJOB
3634  D8B4  48                    PHA              ;CLR JOBRTN FLAG
3635  D8B5  A9 00                 LDA #0
3636  D8B7  8D 9A 43              STA JOBRTN
3637  D8BA  68                    PLA
3638  D8BB  60                    RTS
3639  D8BC               ;
3640  D8BC               ;
3641  D8BC               ;TEST IF JOB(.X) IS DONE YET
3642  D8BC               ;IF NOT DONE RETURN
3643  D8BC               ;IF OK THEN RETURN ELSE REDO IT
3644  D8BC               ;
3645  D8BC  BD 03 10     TSTJOB   LDA JOBS,X
3646  D8BF  30 11                 BMI NOTYET
3647  D8C1  C9 02                 CMP #2
3648  D8C3  90 0B                 BCC OK
3649  D8C5               ;
3650  D8C5  C9 08                 CMP #8           ;CHECK FOR WP SWITCH ON
3651  D8C7  F0 04                 BEQ TJ10
3652  D8C9               ;
3653  D8C9  C9 0B                 CMP #11          ;CHECK FOR ID MISMATCH
3654  D8CB  D0 07                 BNE RECOV
3655  D8CD               TJ10
3656  D8CD  4C 48 D9              JMP QUIT2
3657  D8D0               ;
3658  D8D0  18           OK       CLC                 ;C=0 FINISHED OK OR QUIT
3659  D8D1  60                    RTS
3660  D8D2               ;
3661  D8D2  38           NOTYET   SEC                 ;C=1 NOT YET
3662  D8D3  60                    RTS
3663  D8D4               ;
3664  D8D4               ;
3665  D8D4               RECOV
3666  D8D4  98                    TYA              ;SAVE .Y
3667  D8D5  48                    PHA
3668  D8D6  A5 12                 LDA DRVNUM       ;SAVE DRIVE #
3669  D8D8  48                    PHA
3670  D8D9  20 B3 D9              JSR DOREC
```

```
LINE# LOC   CODE        LINE


3671  D8DC  C9 02                 CMP #2
3672  D8DE  B0 03                 BCS REC01
3673  D8E0  4C 79 D9              JMP REC95
3674  D8E3              REC01
3675  D8E3  BD 50 43              LDA LSTJOB,X
3676  D8E6  48                    PHA              ;SAVE ORIGINAL JOB FOR LATER
3677  D8E7  29 01                 AND #1
3678  D8E9  85 12                 STA DRVNUM       ;SET ACTIVE DRIVE #
3679  D8EB                                         ;
3680  D8EB  68                    PLA              ;ORIGINAL JOB
3681  D8EC  29 F0                 AND #$F0         ;MASK JOB CODE
3682  D8EE  48                    PHA              ;SAVE IT
3683  D8EF  C9 90                 CMP #WRITE
3684  D8F1  D0 07                 BNE REC0         ;NOT A WRITE
3685  D8F3              ;
3686  D8F3  A5 12                 LDA DRVNUM
3687  D8F5  09 B8                 ORA #SECSEK      ;REPLACE W/ SECTOR SEEK...
3688  D8F7  9D 50 43              STA LSTJOB,X     ;... DURING RECOVERY
3689  D8FA              REC0
3690  D8FA  2C F5 10              BIT REVCNT
3691  D8FD  70 3A                 BVS REC5         ;NO TRACK OFFSET
3692  D8FF              ;
3693  D8FF  A9 00                 LDA #0
3694  D901  8D 9B 43              STA EPTR         ;CLR OFFSET TABLE PTR
3695  D904  8D 9C 43              STA TOFF         ;CLR TOTAL OFFSET
3696  D907              REC1
3697  D907  AC 9B 43              LDY EPTR
3698  D90A  AD 9C 43              LDA TOFF
3699  D90D  38                    SEC
3700  D90E  F9 B8 10              SBC OFFSET,Y
3701  D911  8D 9C 43              STA TOFF         ;KEEP TRACK OF ALL OFFSETS
3702  D914  B9 B8 10              LDA OFFSET,Y
3703  D917  20 83 D9              JSR HEDOFF
3704  D91A              ;
3705  D91A  EE 9B 43              INC EPTR         ;BUMP TABLE PTR
3706  D91D  20 B3 D9              JSR DOREC        ;DO THE RECOVERY
3707  D920  C9 02                 CMP #2           ;ERROR CODE < 2?
3708  D922  90 08                 BCC REC3         ;JOB WORKED
3709  D924              ;
3710  D924  AC 9B 43              LDY EPTR
3711  D927  B9 B8 10              LDA OFFSET,Y
3712  D92A  D0 DB                 BNE REC1         ;NULL INDICATES END
3713  D92C              REC3
3714  D92C  AD 9C 43              LDA TOFF
3715  D92F  20 83 D9              JSR HEDOFF
3716  D932              ;
3717  D932  BD 03 10              LDA JOBS,X
3718  D935  C9 02                 CMP #2
3719  D937  90 2F                 BCC REC9         ;NO ERROR
3720  D939              REC5
3721  D939  2C F5 10              BIT REVCNT       ;CHECK BUMP-ON FLAG
3722  D93C  10 10                 BPL REC7         ;NO BUMP
3723  D93E              ;
3724  D93E              QUIT
3725  D93E  68                    PLA
```

```
LINE# LOC   CODE        LINE

3726  D93F  C9 90                 CMP #WRITE     ;CHECK ORIGINAL JOB
3727  D941  D0 05                 BNE QUIT2
3728  D943              ;
3729  D943  05 12                 ORA DRVNUM
3730  D945  9D 50 43              STA LSTJOB,X   ;MUST RESTORE ORIGINAL
3731  D948              QUIT2
3732  D948  BD 03 10              LDA JOBS,X     ;.A= ERROR #
3733  D94B  20 B3 EE              JSR ERROR
3734  D94E              REC7
3735  D94E  68                    PLA
3736  D94F  2C 9A 43              BIT JOBRTN
3737  D952  30 25                 BMI REC95
3738  D954  48                    PHA
3739  D955                                       ; DO THE BUMP
3740  D955  A9 C0                 LDA #BUMP
3741  D957  05 12                 ORA DRVNUM
3742  D959  9D 03 10              STA JOBS,X
3743  D95C              REC8
3744  D95C  BD 03 10              LDA JOBS,X
3745  D95F  30 FB                 BMI REC8       ;WAIT
3746  D961              ;
3747  D961  20 B3 D9              JSR DOREC      ;TRY ONE LAST SET
3748  D964  C9 02                 CMP #2
3749  D966  B0 D6                 BCS QUIT       ;IT CLEARLY AIN'T GONNA WORK
3750  D968              REC9
3751  D968  68                    PLA            ;CHECK ORIGINAL JOB FOR WRIT
3752  D969  C9 90                 CMP #WRITE
3753  D96B  D0 0C                 BNE REC95      ;ORIGINAL JOB WORKED
3754  D96D              ;
3755  D96D  05 12                 ORA DRVNUM
3756  D96F  9D 50 43              STA LSTJOB,X   ;SET WRITE JOB BACK
3757  D972  20 B3 D9              JSR DOREC      ;TRY LAST SET OF WRITES
3758  D975  C9 02                 CMP #2         ;CHECK ERROR CODE
3759  D977  B0 CF                 BCS QUIT2      ;ERROR
3760  D979              REC95
3761  D979  68                    PLA
3762  D97A  85 12                 STA DRVNUM     ;RESTORE DRIVE #
3763  D97C  68                    PLA
3764  D97D  A8                    TAY            ;RESTORE .Y
3765  D97E  BD 03 10              LDA JOBS,X
3766  D981  18                    CLC            ;OK!
3767  D982  60                    RTS
3768  D983              ;
3769  D983              HEDOFF                   ;.A=OFFSET
3770  D983  C9 00                 CMP #0
3771  D985  F0 18                 BEQ HOF3       ;NO OFFSET
3772  D987  30 0C                 BMI HOF2       ;STEPS ARE INWARD
3773  D989              HOF1
3774  D989  A0 01                 LDY #1         ;STEP OUT 1 TRACK
3775  D98B  20 A0 D9              JSR MOVHED
3776  D98E  38                    SEC
3777  D98F  E9 01                 SBC #1
3778  D991  D0 F6                 BNE HOF1       ;NOT FINISHED
3779  D993  F0 0A                 BEQ HOF3
3780  D995              HOF2
```

```
LOC   CODE        LINE

D995  A0 FF                 LDY #$FF        ;STEP IN 1 TRACK
D997  20 A0 D9              JSR MOVHED
D99A  18                    CLC
D99B  69 01                 ADC #1
D99D  D0 F6                 BNE HOF2        ; NOT FINISHED
D99F              HOF3
D99F  60                    RTS
D9A0              ;
D9A0              MOVHED
D9A0  48                    PHA             ;SAVE .A
D9A1  98                    TYA             ;PUT PHASE IN .A
D9A2  A4 12                 LDY DRVNUM
D9A4  99 A1 10              STA PHASE,Y
D9A7              MH10
D9A7  D9 A1 10              CMP PHASE,Y
D9AA  F0 FB                 BEQ MH10        ;WAIT FOR CONTROLLER
D9AC                                       ;TO CHANGE IT
D9AC  A9 00                 LDA #0
D9AE  99 A1 10              STA PHASE,Y     ;CLEAR IT OUT
D9B1  68                    PLA             ;RESTORE
D9B2  60                    RTS
D9B3              ;
D9B3              ;
D9B3              DOREC                      ;DO LAST JOB RECOVERY
D9B3  AD F5 10              LDA REVCNT      ;RE-TRY JOB REVCNT...
D9B6  29 3F                 AND #$3F        ;...# OF TIMES
D9B8  A8                    TAY
D9B9              DOREC1
D9B9  AD 82 02              LDA PBD2
D9BC  49 20                 EOR #ERRLED
D9BE  8D 82 02              STA PBD2
D9C1  BD 50 43              LDA LSTJOB,X    ;SET LAST JOB
D9C4  9D 03 10              STA JOBS,X
D9C7              DOREC2
D9C7  BD 03 10              LDA JOBS,X      ;WAIT
D9CA  30 FB                 BMI DOREC2
D9CC  C9 02                 CMP #2
D9CE  90 03                 BCC DOREC3      ;IT WORKED
D9D0              ;
D9D0  88                    DEY
D9D1  D0 E6                 BNE DOREC1      ;KEEP TRYING
D9D3              DOREC3
D9D3  48                    PHA
D9D4  20 F2 F9              JSR ERROFF
D9D7  68                    PLA
D9D8  60                    RTS             ;FINISHED
D9D9              ;
D9D9              ; SET HEADER OF ACTIVE BUFFER OF THE
D9D9              ; CURRENT LINDX TO TRACK,SECTOR,ID
D9D9              ;
D9D9  20 D5 E7    SETHDR JSR GETACT
D9DC              SETH
D9DC  0A                    ASL A
D9DD  0A                    ASL A
D9DE  0A                    ASL A
```

```
LINE# LOC   CODE        LINE

3836  D9DF  A8                 TAY
3837  D9E0  A5 13              LDA TRACK
3838  D9E2  99 23 10           STA HDRS+2,Y    ;SET TRACK
3839  D9E5  A5 14              LDA SECTOR
3840  D9E7  99 24 10           STA HDRS+3,Y    ;SET SECTOR
3841  D9EA  A5 12              LDA DRVNUM      ;GET PROPER ID(DRVNUM)
3842  D9EC  0A                 ASL A
3843  D9ED  AA                 TAX
3844  D9EE  BD 42 43           LDA DSKID,X
3845  D9F1  99 21 10           STA HDRS,Y
3846  D9F4  BD 43 43           LDA DSKID+1,X
3847  D9F7  99 22 10           STA HDRS+1,Y
3848  D9FA  60                 RTS
3849  D9FB              ;
3850  D9FB                     .END
3851  D9FB                     .LIB ADDFIL
```

```
LINE# LOC   CODE        LINE


2907  D47F              ;
2908  D47F              ;
2909  D47F              ;*
2910  D47F              ;*
2911  D47F              ;******************************
2912  D47F              ;*
2913  D47F              ;*   FNDWCH
2914  D47F              ;*
2915  D47F              ;******************************
2916  D47F              ;*
2917  D47F              ;*
2918  D47F  A5 16       FNDWCH LDA SA
2919  D481  C9 13              CMP #MAXSA+1
2920  D483  90 02              BCC FNDW13
2921  D485  29 0F              AND #$F
2922  D487  AA          FNDW13 TAX
2923  D488  B5 AC              LDA LINTAB,X
2924  D48A  A8                 TAY
2925  D48B  0A                 ASL A
2926  D48C  90 0A              BCC FNDW15
2927  D48E  30 0A              BMI FNDW20
2928  D490  98          FNDW10 TYA
2929  D491  29 0F              AND #$0F
2930  D493  85 15              STA LINDX
2931  D495  AA                 TAX
2932  D496  18                 CLC
2933  D497  60                 RTS
2934  D498  30 F6       FNDW15 BMI FNDW10
2935  D49A  38          FNDW20 SEC
2936  D49B  60                 RTS
2937  D49C              ;
2938  D49C              TYPFIL                    ;GET FILE TYPE
2939  D49C  A6 15              LDX LINDX
2940  D49E  B5 9A              LDA FILTYP,X
2941  D4A0  4A                 LSR A
2942  D4A1  29 07              AND #7
2943  D4A3  C9 04              CMP #RELTYP
2944  D4A5  60                 RTS
2945  D4A6              ;
2946  D4A6  20 D5 E7    GETPRE JSR GETACT
2947  D4A9  0A                 ASL A
2948  D4AA  AA                 TAX
2949  D4AB  A4 15              LDY LINDX
2950  D4AD  6C                 RTS

2952  D4AE              ; READ BYTE FROM ACTIVE BUFFER
2953  D4AE              ; AND SET FLAG IF LAST DATA BYTE
2954  D4AE              ; IF LAST THEN Z=1 ELSE Z=0 ;
2955  D4AE  20 A6 D4    GETBYT JSR GETPRE
2956  D4B1  B9 C7 00           LDA LSTCHR,Y
2957  D4B4  F0 12              BEQ GETB1
2958  D4B6  A1 29              LDA (BUFTAB,X)
2959  D4B8  48                 PHA
2960  D4B9  B5 29              LDA BUFTAB,X
2961  D4BB  D9 C7 00           CMP LSTCHR,Y
```